Chapter 8
Supporting Infrastructure

A principal tenet of the Defense-in-Depth philosophy is to provide defenses against cyber intrusions and attacks, and deal effectively with and recover from attacks that penetrate those defenses. The supporting infrastructures are a set of interrelated activities and infrastructures providing security services to enable and manage the framework's technology solutions. Currently, the Defense-in-Depth strategy defines two supporting infrastructures:

- **Key Management Infrastructure/Public Key Infrastructure (KMI/PKI).** For the generation, distribution, and management of security credentials, such as keys and certificates.
- **Detect and Respond.** For providing warnings, detecting and characterizing suspected cyber attacks, coordinating effective responses, and performing investigative analyses of attacks.

Today's information infrastructures are not sufficiently secure to provide the full range of services needed to defend against the threats anticipated for the Global Information Grid (GIG). Thus, the Defense-in-Depth strategy provides overlays of information assurance (IA) features to realize an effective defense. Key management (including public key management) is fundamental to many IA protection technologies. Because our ability to provide airtight protection is neither technically nor economically feasible, we must reinforce those protection technologies with capabilities to detect, respond to, and recover from cyber attacks that penetrate those protections.

Cryptography-enabled services rely on KMI or PKI to provide a trustworthy foundation. The KMI/PKI supporting infrastructure focuses on the technologies, services, and processes used to manage public key certificates and symmetric cryptography. As shown in Figure 8-1, the KMI/PKI infrastructure touches most portions of the networked environment.

KMI/PKI hardware and software at the enclave level provide local authorities (e.g., KMI managers) with capabilities to order and manage KMI/PKI products and services, issue certificates, and generate traditional symmetric keys. KMI at the wide area network (WAN) level provides certificate, directory, and key generation and distribution functions.

The PKI strategy is based heavily on multiple levels of assurance because it is not cost effective to provide high-assurance protection for all PKI-enabled services. High assurance is needed when public key capabilities are used as the primary means to protect national security information. For other services, a medium-assurance PKI is appropriate based on commercial technology. The medium-assurance PKI will initially use software-based end-user tokens, but it will evolve to the use of hardware tokens.

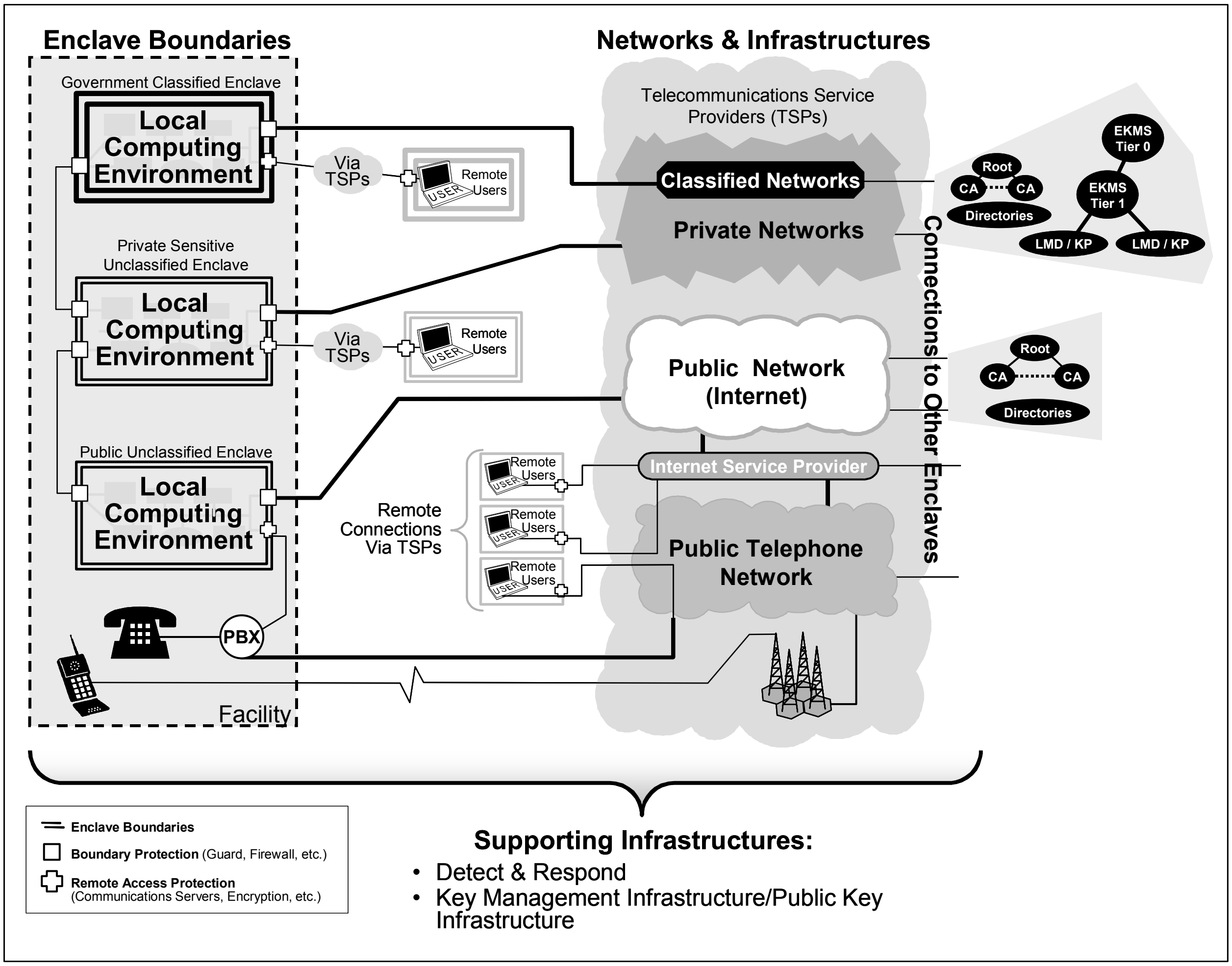

Figure 8-1. Supporting Infrastructures: KMI/PKI

Because a major feature of a PKI is to provide widespread interoperability and a broad base of noninteroperable commercial PKI technology solutions exists on the market today, we recommend a foundational PKI be fielded quickly so that other efforts can build on it. The PKI should support interoperability with external federal, foreign, and public domains. One way to achieve interoperability is through cross-certification. Further study is required to decide where cross-certification is best used. With PKI technology still immature and changing rapidly, the strategy for fielding a large-scale PKI quickly should be to make it a simple infrastructure that provides only basic cryptographic capabilities, including digital identifications (ID), compromise recovery, key recovery, and archive. Departments, agencies, and corporations are then free to build atop this infrastructure for capabilities such as access control.

It is unclear whether the higher assurance PKI is best operated by corporate personnel or outsourced. Numerous government organizations (including a major effort by the Department of Defense [DoD]) deploy and operate PKI pilots to gather operating information to evaluate its impact on mission (and business) performance and assess whether portions should be outsourced.

The local environments will maintain the option of deploying sensors, and possibly analysts to interpret the results of, and, when appropriate, react to the implications of these outputs. Beyond the local environment, each organization, or perhaps community, must determine what information should be reported, in what format, under what situations, and to whom.

While planning for a Detect and Respond infrastructure, it is important to recognize that the enterprise networks and systems that it will support must also be structured to provide information to, and take advantage of, the services and information that the infrastructure provides. This section provides good engineering practices for an enterprise to enhance its Detect and Respond capability.

When considering a general construct for a Detect and Respond infrastructure, a primary consideration is the perspective that the infrastructure will provide for its support. The reality is that most infrastructures are inherently hierarchical, and this one is no exception. Often information about incidents, which is usually sensed at the lowest layer in the hierarchy, is promulgated up to higher layers with some form of reporting. Warning and response coordination that are more typically derived from higher layers are disseminated from those higher layers down.

A wide range of functions is needed to support Detect and Respond, and technology solutions are not available to automatically perform many of these functions. Thus, analysts, network operators, and system administrators who apply basic support technologies to ease their tasks perform many of these functions. To deal with this issue from a technology viewpoint, we identify the functions that these analysts (and their tools) are attempting to perform, and then discuss the technologies that are available to realize these functions.

The Detect and Respond infrastructure element provides the functional and management capabilities to provide warning alerts of possible upcoming cyber attacks, and to assist local environments to detect, characterize, respond to, and recover from attacks. Figure 8-2 highlights the areas of the high-level Defense Information Infrastructure (DII) context that comprise the detect and respond infrastructure.

Because the local environments are the logical location for sensors, the network-based sensor functions are discussed in Chapter 6, Defend the Enclave Boundary/External Connections, and their host-based counterparts are covered in Chapter 7, Defend the Computing Environment. We recognize that local environments have the option to implement as much or as little as they believe is prudent, obtaining services and support from the infrastructure. Detect and Respond processes and functions in the context of the supporting infrastructure are the focus of this section.

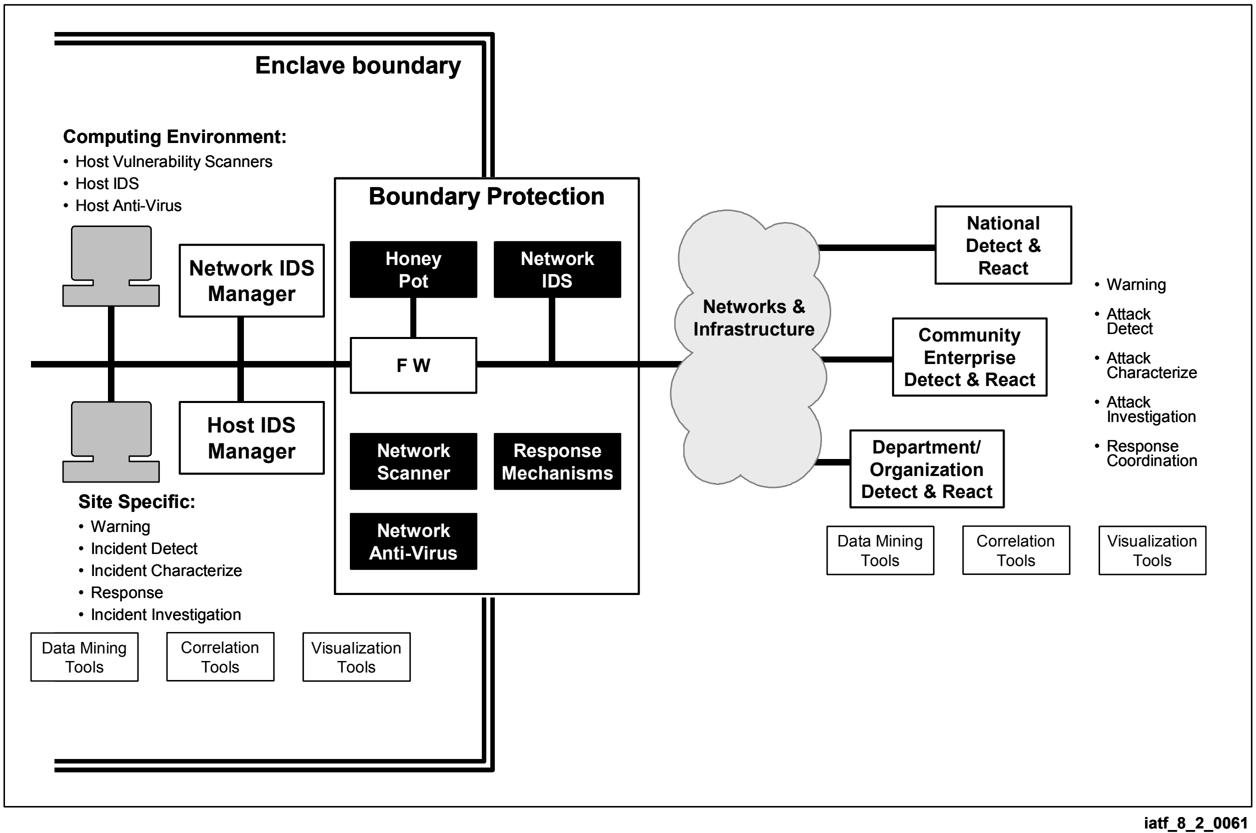

Figure 8-2. Supporting Infrastructures: Detect and Respond

8.1 Key Management Infrastructure/ Public Key Infrastructure

This section focuses on management of the Supporting infrastructure. Following introductory tutorial information, Public Key Infrastructure (PKI) certificate management, symmetrical key management, directory management, and infrastructure management will be highlighted. Each of the process discussions is self-contained; therefore, the reader can review only those Key Management Infrastructure (KMI)/PKI services and processes that are of interest. They will include specific requirements applicable to that process and KMI/PKI service, important threats and countermeasures, and the range of technologies used to implement the process. Table 8.1-3 defines at a high level, the way each process relates to the various KMI/PKI services. The remainder of the section presents a range of KMI/PKI solutions used by or planned for protected networks.

8.1.1 KMI/PKI Introduction

KMI/PKI is unique in its framework because it does not directly satisfy subscriber's security requirements; instead, it forms building blocks used by other security technologies. The KMI/PKI is an enabler; however, the KMI/PKI architecture is heavily dependent on the specific applications it supports. Table 8.1-1 relates the subscriber categories described in Chapters 5, Defend the Network and Infrastructure and 6, Defend the Enclave Boundary/External Connections, of the framework and to the required KMI/PKI services. For example, a virtual private network (VPN) provides an encrypted pipe between two enclaves. The KMI/PKI infrastructure supplies keys and certificates to the cryptographic devices that provide authentication and encryption. Additional services might include key recovery and a directory to provide access to subscriber's public certificates.

Table 8.1-1. KMI/PKI Services Support to Subscriber Categories

Subscriber Category	KMI/PKI Service			
VPN	Key generation	Certificate management	Key recovery	Directory
Network Access	Key generation	Certificate management	Value-added services	Directory
Remote Access	Key generation	Certificate management	Key recovery	Directory
Multilevel Security	Key generation	Certificate management	Directory	

Another area in which KMI/PKI differs from the Framework's other solutions is that it distributes its security throughout a number of separate elements. These elements require extensive security (e.g., encryption, certificate management, compromise recovery) among themselves to protect the subscriber's key or certificate. Because of the repercussions of a successful attack against the KMI/PKI, internal infrastructure security requirements are often more stringent than those required by subscriber applications. There are unique requirements on the infrastructure (e.g., policy management) and the level of security assurance for infrastructure components is usually higher than for subscriber applications.

8.1.1.1 KMI/PKI Services

Current KMI/PKI implementations consist of several stovepipe infrastructures from different organizations, supplying different subscriber solutions. The end subscriber may need support from several of the stovepipes for a single application. Today, subscribers have to contact each infrastructure separately to get service. High cost, dwindling manpower, and higher subscriber expectations are pressuring a merger of these stovepipes into larger infrastructure elements supporting multiple subscriber requirements.

This chapter discusses four of the operational services supplied by the KMI/PKI supporting infrastructure. These KMI/PKI services support many subscriber applications and consequently employ different (but related) mechanisms and have unique security requirements. The first two services describe functions that directly support subscriber applications. The last two services are functions required by the subscriber functions to work properly.

The first KMI/PKI service is symmetric key generation and distribution. This is still the primary key management mechanism within the government classified community. The banking community, with its extensive use of the data encryption standard (DES) encryption, is another major user of symmetric key management. Although symmetric key is being replaced by asymmetric key agreement in many applications, it has application outside the government classified community in such areas as multicast and low-bandwidth applications (e.g., wireless). Symmetric key management is a process in which a central element (it could be one of the subscribers or a trusted independent element) generates, distributes, and manages a “secret key” for multiple recipients. Each recipient uses the same secret key for security processing between itself and the other recipients for the life of the key.

The second KMI/PKI service is support for asymmetric cryptography *(*often called public key cryptography) and its associated certificate management. Asymmetric cryptography usually employs digital certificates to allow subscribers to authenticate the public portion of the asymmetric cryptography public and private key pairs. This authentication is important because the security services that asymmetric cryptography provides depend on the subscriber of a public key (called the relying party) being assured that the public key is associated with a specific identified subscriber. Digital certificates (often called X.509 certificates, after the international standard which defines their format) cryptographically bind identities to public keys. Together, the components, personnel, facilities, services, and policies that are used to generate and manage public key certificates define a PKI. PKIs can generate and manage digital signature certificates (used for authentication, data integrity, and nonrepudiation) and key management certificates (used for confidentiality). The commercial community relies heavily on public key cryptography, and commercial vendors offer a wide variety of PKI products and services.

The third KMI/PKI service is directory service. Directory servers provide access to the public information required with PKI, such as the public certificate, the related infrastructure certificates, and the compromised key information. Directory services can be provided either by a global set of distributed directories (e.g., X.500 Defense Message System [DMS] directories) or by an online repository at a single site. Directories are normally very closely coupled with PKI, but are also used for other services.

The final KMI/PKI service is managing the infrastructure itself. The other infrastructure architectures discussed in this section consist of a number of elements working together to provide the subscriber service. The distributed nature of the infrastructure places additional functional and procedural requirements on the KMI/PKI and the sensitivity of the application places additional security requirements on the KMI/PKI. The internal structure of the infrastructure varies with the application(s) it supports. For example, the level of assurance demanded by the applications dictates many of the internal aspects of the KMI/PKI.

8.1.1.2 Security Applications

The security applications supported by the KMI/PKI differ depending on the type of cryptography that is being used by the application. Symmetric cryptography primarily provides confidentiality services for data transmission and storage. It can also support other mechanisms such as transmission security (TRANSEC) (e.g., spread spectrum), or in combination with additional mechanisms, data integrity, and authentication during data transmission. Public key cryptography in conjunction with certificate management provides a full range of security services. Unlike symmetric cryptography, it can provide authentication and integrity for data transmission and data storage. Although it can encrypt information, this process is extremely inefficient and is normally provided by a symmetric algorithm. Table 8.1-2 describes the security applications that each type of cryptographic algorithm supports.

Table 8.1-2. Security Applications Supported By Cryptographic Type

Security Application	Symmetric Cryptography	Asymmetric Cryptography
Authentication	*	X
Nonrepudiation	*	X
Transmission Confidentiality	X	
File Encryption	X	
Integrity	*	X
Availability (e.g., Spread Spectrum)	X	
Key Agreement		X

*These services can be enabled by symmetric cryptography when provided in conjunction with other mechanisms (e.g., a cyclic redundancy check [CRC] encrypted with the message).

8.1.1.3 Infrastructure Process

The KMI/PKI consists of numerous processes that all have to work together correctly for the subscriber service to be secure. Each process is necessary at some level in all KMI/PKI architectures. These processes are listed below:

- **Registration**—Enrolling those individuals who are authorized to use the KMI/PKI.
- **Ordering**—Requesting the KMI/PKI to provide a subscriber either a key or a certificate.
- **Key Generation**—Generating the symmetric or asymmetric key by an infrastructure element.
- **Certificate Generation**—Binding the subscriber information and the asymmetric key into a certificate.
- **Distribution**—Providing the keys and certificates to the subscribers in a secure, authenticated manner.
- **Accounting**—Tracking the location and status of keys and certificates.
- **Compromise Recovery**—Removing compromised keys and invalid certificates from the system in an authenticated manner.
- **Rekey**—Replacing periodically the keys and certificates in a secure, authenticated manner.
- **Destruction**—Destroying the secret key when it is no longer valid.
- **Key Recovery**—Recovering subscriber's private encryption key without direct access to the subscriber's copy of the key.
- **Policy Creation**—Defining the requirements for employment of the previous processes.
- **Administration**—Running the infrastructure.
- **Value-added PKI Processes**—Supporting optional value-added processes, including archive, time stamp, and notary services. Because all PKI architectures do not support these features, this section will not discuss them further.

The complete set of KMI/PKI processes is usually allocated to several elements performing independent tasks that require extensive coordination between elements. For most of the processes, numerous ways exist to implement the services based on the application supported; the security required; and the cost (e.g., money, people, and performance) the subscriber would be willing to pay. Each process contributes to the overall security of the KMI/PKI and has different forms of attacks and countermeasures. Table 8.1-3 defines the basic requirements for implementing each process for the four KMI/PKI services. Figure 8.1-1 depicts the interaction of these services.

Table 8.1-3. KMI/PKI Processes

Process	Certificate (Public Key) Management, Section 8.1.2	Symmetric Key Management, Section 8.1.3	Infrastructure Directory Services, Section 8.1.4	Infrastructure Management, Section 8.1.5
Policy Creation	N/A	N/A	N/A	Define domain's policy and method for enforcing the policy
Registration	Register people who can authorize subscribers	Register people authorized to order key	Register people authorized to update directory	Define process of authorizing changes to the infrastructure's trust model (e.g., new elements, cross-certification)
Ordering and Validation	• Validate the information in the certificate • Validate the key generation request • Receive the public key	Validate order	Validate the information request	• Validate process for changes to the trust model • Receive the public key of the infrastructure elements
Generation	• Generate the public/private key pairs • Generate the certificate	Generate key	Add information to the directory	• Generate the root public/private keys • Generate the root certificate • Generate the infrastructure elements public/private keys • Generate the infrastructure elements certificates • Generate the cross-certificates
Distribution	• Provide the certificate to the subscriber • Validate that the person getting the certificate has the private key corresponding to the bound public key • Provide the Policy Approving Authority (PAA) public certificate to the subscriber in an authenticated manner	• Deliver the key to the Custodian • Load the key into the cryptograp hic device	Provide information to subscriber	• Provide the root certificate to each infrastructure element in an authenticated manner • Provide each element with its certificates • Validate that each infrastructure element has the private key corresponding to the public key • Provide each element with the domain's cryptographic parameters in an authenticated manner

Process	Certificate (Public Key) Management, Section 8.1.2	Symmetric Key Management, Section 8.1.3	Infrastructure Directory Services, Section 8.1.4	Infrastructure Management, Section 8.1.5
Compromise Recovery	• Provide Compromise Key List (CKL) of compromised keys • Provide online validation of the liveness of certificates	Provide supersession of all devices using the compromised key	Fix a hacked directory	Provide procedures for reconstituting the infrastructure in case of disaster or compromise of any infrastructure element
Accounting	Track the location and status of key and certificates throughout life cycle	Track the location and status of key throughout life-cycle	Audit who makes changes to the information in the directory	Ensure that the infrastructure elements operate within the policies and procedures defined by the PAA
Key Recovery	Appropriate key recovery mechanisms	N/A	N/A	Root signature key might need key recovery?
Rekey	• New certificate • New key	Rekey the cryptographic device	N/A	Process for changing the root key(s)
Destruction	Zeroize private key at the conclusion of use	Zeroize key at the conclusion of the cryptoperiod	Remove information from the directory	Zeroize the infrastructure elements private key at the conclusion of use
Administration	N/A	N/A	N/A	Provide procedures for operating the infrastructure securely and enforcing the system policies

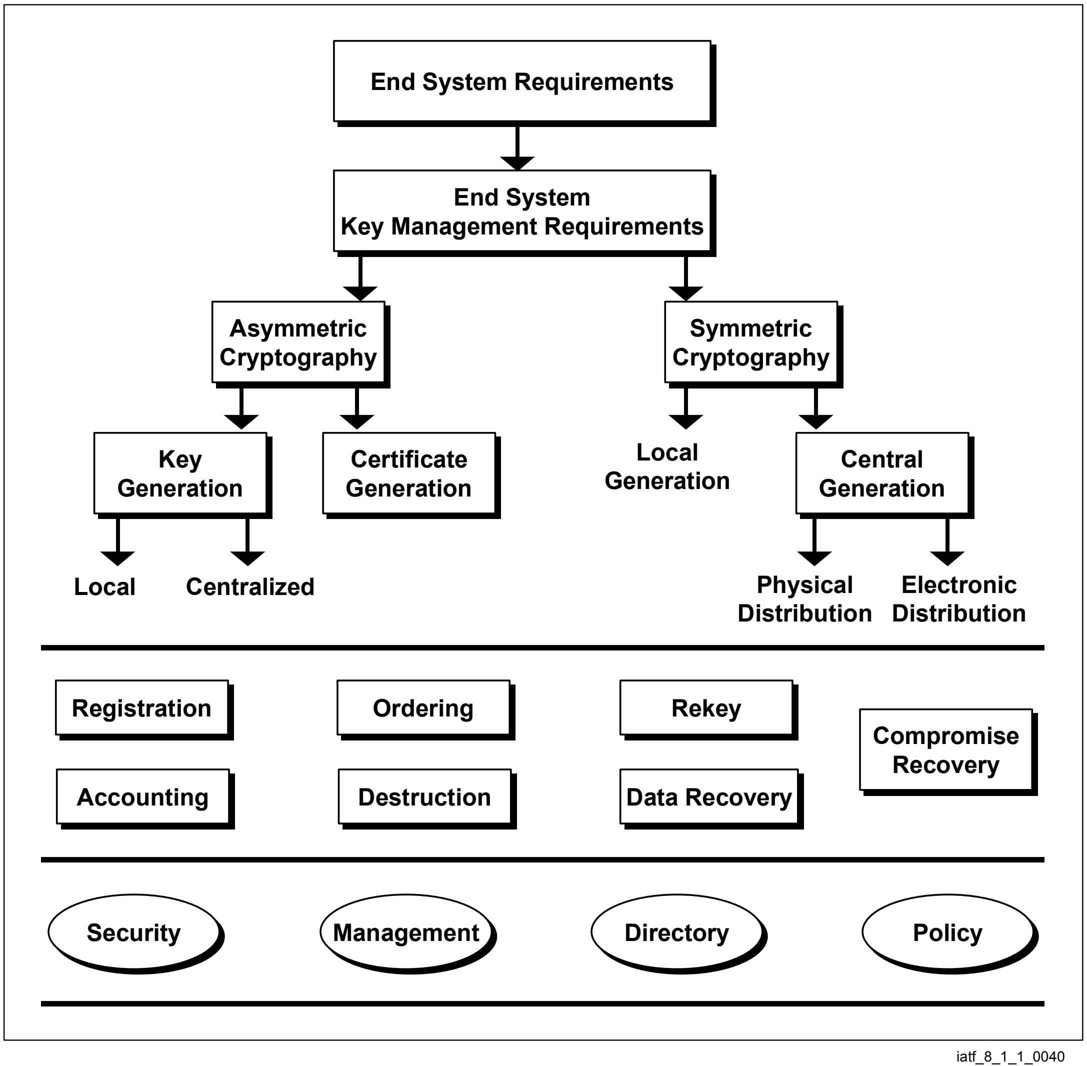

Figure 8.1-1. Interactions of the KMI/PKI Applications Operational Services

8.1.1.4 Requirements

This section includes subscriber and infrastructure requirements. Because of the variety of issues involved in KMI/PKI, no single set of requirements can be consistent and complete for all applications. This paragraph outlines some of the high-level requirements. It consists of both functional and operational requirements. Unlike most of the subscriber requirements identified in the Framework, the KMI/PKI has a large operational component. Once initialized, most subscriber solutions need little or no subscriber interaction (e.g., once the VPN has deployed the cryptographic device, the only update is the KMI/PKI task of rekeying periodically). The KMI/PKI, on the other hand, requires extensive human interaction throughout its processing. This close coupling of people and service place additional requirements on the KMI/PKI that have implications on the security solution.

8.1.1.4.1 Subscriber Requirements

Symmetric Cryptography

- The key comes from an approved, authorized, authenticated source.
- The key is protected during distribution.

Asymmetric Cryptography

- The subscriber or the KMI/PKI shall generate the public and private key pair.
- The certificate information is accurate and current, and it reflects a valid association with a uniquely identified subscriber.
- The certificate binds the public key associated with the subscriber’s private key with the subscriber’s identification.
- The trusted element’s certificate is distributed to the subscriber in an authenticated manner.
- The subscriber can determine the current status of certificates in a timely manner.
- The KMI/PKI only provides a copy of a private key to authorize data recovery entities as defined by policy (e.g., subscriber or subscriber’s organization).

8.1.1.4.2 Infrastructure Management Requirements

Symmetric Cryptography

- Symmetric cryptography ensures that requests for key generation or distribution come from only authorized sources.
- Key generation is secure and robust.
- The delivery mechanism protects the key from compromise.
- Key is distributed to only authorized subscribers.
- The system accounts for key during its entire life cycle (ordering, generation, distribution, use, rekey, and destruction).
- The infrastructure removes compromised keys from the system.

Asymmetric Cryptography

- Asymmetric cryptography ensures that a request for a certificate comes from an authorized source.
- Before generating the certificate, the system ensures that the information in the certificate corresponds to the requesting subscriber.
- The certification authority (CA) places the correct public key into the certificate.

- If the infrastructure generates the private key agreement key, it is generated and transmitted securely to the subscriber.
- The infrastructure must ensure integrity and provide its certificates in an authenticated nonrepudiated manner to each subscriber.
- The infrastructure must provide compromise information to subscribers in a timely manner.
- The infrastructure must ensure high assurance in the registration of infrastructure elements.
- The system accounts for the life cycle of key (ordering, generation, distribution, application, rekey, destruction, and archive).
- The key recovery mechanism of the KMI/PKI only provides access to the private key to authorized entities (e.g., subscriber's organization).
- The key must be protected by the key recovery mechanism of the KMI/PKI during storage.
- The recovered key must be protected during distribution to the subscriber.

8.1.1.4.3 Interoperability Requirements

NOTE: Interoperability of the key management cryptographic infrastructure does not guarantee subscriber application interoperability.

Symmetric Cryptography

- Keys and compromise information can be distributed to all subscribers.
- Format of the key must be the same for all subscribers.
- Algorithms and initial parameters must be the same for all subscribers.

Asymmetric Cryptography

- When cross-certifying, the policies must be approved by each PKI.
- The subscriber may need to accept certificates from multiple domains.
- The infrastructure may need to support multiple algorithms and offer the subscriber the choice of algorithm to sign the certificate.
- The format of the keys and certificates must be the same for all subscribers (e.g., certificate profiles, use of X.509).
- Algorithms and initial parameters must be the same for all subscribers.
- Compromise recovery information must be available to all subscribers.

8.1.1.5 Attacks and Countermeasures

The goal of any attack against the infrastructure is to use it as a basis for attacking a subscriber's environment. Attacking the infrastructure does not provide an adversary with the subscriber's information (beyond audit information that may be archived), but it may be used as a basis for a further attack against the subscriber. An attacker may directly target the information provided by the infrastructure (e.g., symmetric key, certificate) or may attack the infrastructure elements in order to later attack a subscriber (e.g., place a Trojan horse in an infrastructure element to substitute a known key for the subscriber's valid key). Table 8.1-4 lists several interesting attacks and potential countermeasures.

Table 8.1-4. Attacks and Countermeasures

Attacks Against User Via Infrastructure Support	Attacks Against Infrastructure	Countermeasures
Compromise data [Read traffic resulting from weak cryptography (compromised, weak keys)] Masquerade (get a certificate with false information) Denial of service (prevent signature from verifying [e.g., attack directories]) Man-in-the-middle attack	Violate trust model (e.g., generate an unauthorized cross-certification_ Acquire unauthorized certificate (e.g., insider, incorrect identification) Force subscriber to have weak key (e.g., known key, failed randomizer) Deny by —attacking directories (denial of service) Compromise key during distribution Gain unauthorized access to key recovery key Compromise personal identification number (PIN) to gain access to subscribers private key (generation, distribution, use) Prevent subscriber from determining compromise status during validation Substitute the attacker's public key for the subscriber's public key Place malicious software into infrastructure elements Wage cryptanalytic attack against the PKI's private keys	Use security features of the protocols (e.g., name constraints, policy mapping) Provide proper management of the infrastructure Provide multiperson control on the certificate approval and generation process Provide protected distribution (e.g., benign fill) Provide robust compromise recovery Use tokens to generate and protect private keys Require high-assurance operating systems in infrastructure components Require strong authentication on infrastructure services (e.g., directories and key recovery) Coordinate certificate request content with the security officer, personnel officer, authorization officer, and privilege assignment officer Independently certify the content of certificates against the officially approved certificate requests

8.1.2 Certificate Management

A primary function of KMI/PKIs is the generation, management, and distribution of asymmetric key material and certificates used within a variety of public key-based applications. The portion of the KMI/PKI dedicated to the management of keys and certificates is the PKI. This section provides an overview of the architecture and the processes or functions associated with PKIs. The section also discusses the threats and countermeasures specific to PKIs. This section is written from the perspective of PKI subscribers versus that of PKI administrators. The administrative perspective is discussed in Section 8.1.5.12, Administration.

8.1.2.1 Public Key Infrastructure Services

To support the wide variety of public key-based applications, a PKI employs a diverse set of software and hardware components, protocols, and message formats. The primary components of the PKI are CAs, registration authorities (RA), and certificate repositories. The primary products of the PKI include asymmetric key material, certificates, and Certificate Revocation Lists (CRL). A brief description of these components is provided below.

- **CA.** An authority trusted by one or more subscribers to create and assign certificates. [ISO9594-8] The individual operating the CA equipment is referred to as a CA operator.
- **RA.** A trusted entity responsible for performing tasks, such as authenticating the identity of subscribers requesting certificates on behalf of a CA. The RA neither signs nor issues certificates. Usually, RAs are located at the same location as the subscribers for which they perform authentication. The individual functioning in this role is referred to as the RA operator. Many PKIs distribute the RA functions to Local Registration Authorities (LRA) to provide subscribers with convenient PKI services.
- **Certificate Repository.** The location where a CA posts the certificates and CRLs that it generates so that they are available to PKI subscribers. Repositories can take many forms, including databases and Web servers, but are commonly directories that are accessible using the Lightweight Directory Access Protocol (LDAP).
- **Asymmetric Key Material.** In asymmetric or public key cryptography, two different cryptographic keys are used. One key is used to encrypt or sign data, whereas the other is used to decrypt or verify data. The “private” key is kept secret to the entity generating the key. The “public” key, which is computed from the private key using a mathematical one-way function, is made public. Because it is mathematically infeasible to compute the private key from the public key, knowledge of the public key does not imply knowledge of the private key.
- **Certificates.** A computer-based record that binds a subscriber’s identity (and some authorizations) with his or her public key in a trust association. The certificate identifies the issuing CA, identifies its subscriber, contains the subscriber’s public key, and is digitally signed by the issuing CA. Often, these certificates comply with the International Telecommunications Union (ITU) X.509 standard. Such certificates are called *X.509 certificates*. [1]
- **CRL.** A list containing certificates still within their validity interval, but which no longer represent a valid binding between a public key and a particular identity. CRLs are

created by a CA, and include the certificates revoked by that CA. CRLs may be posted to a repository or may be distributed through another mechanism (e.g., Web and e-mail). Other means for obtaining certificate status, such as Online Certificate Status Protocol, are also sometimes employed instead of CRLs.

Figure 8.1-2 overlays PKI components within a generic security architecture PKI associated with commercial entities, federal partners, and non-federal partners, which are shown along the right-side of the figure. Even though the figure shows federal subscribers obtaining PKI services from federal agency PKIs, federal agencies will often obtain PKI services from commercial providers. Subscribers operating from secure network enclaves (normally, a local area network [LAN] connected to the Internet via a firewall) work with RAs who confirm subscriber identities to obtain certificates from remote CAs.

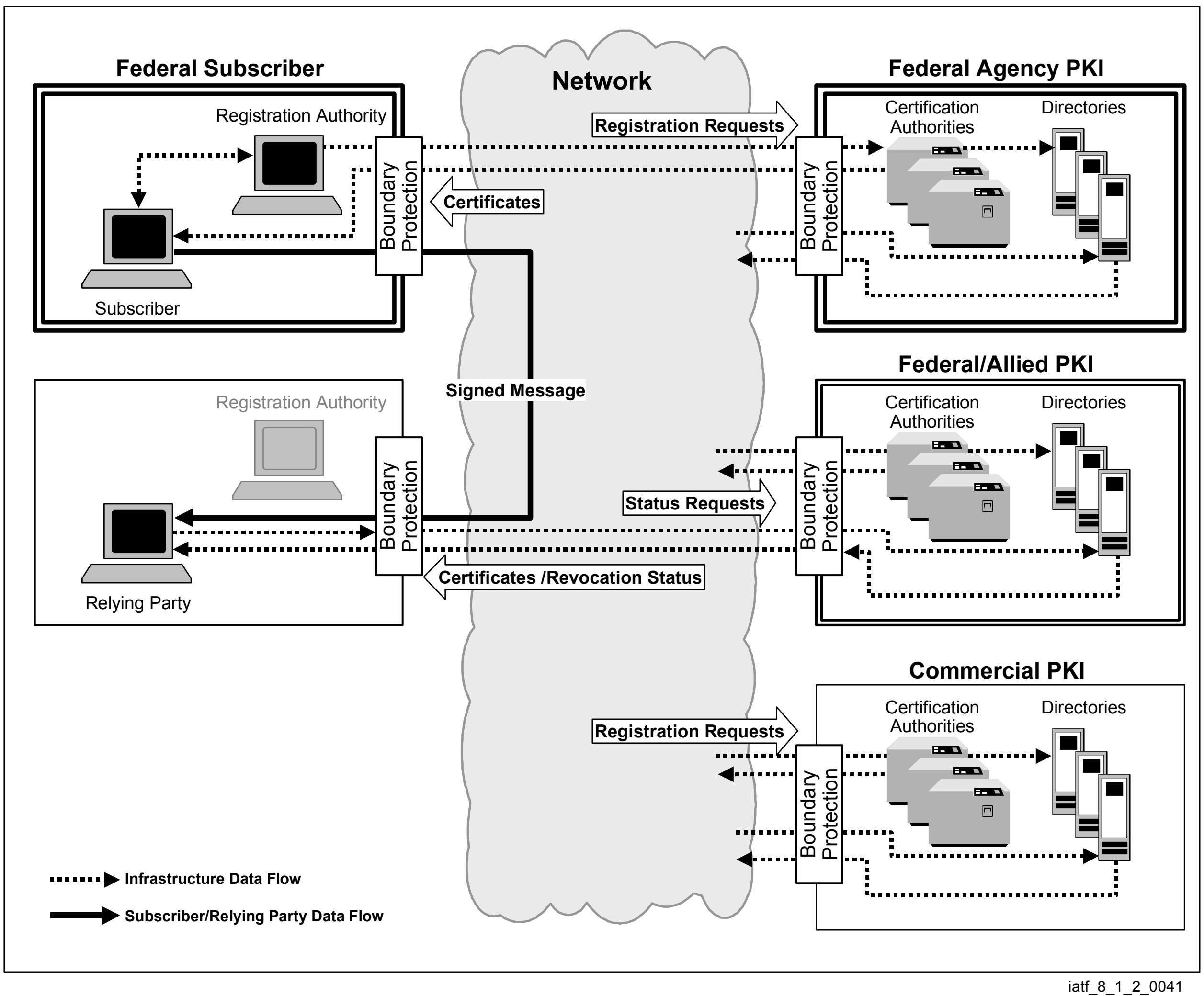

Figure 8.1-2. Using PKIs in Secure Enclaves

Relying parties in other enclaves, associated with other PKIs, may authenticate the subscriber's public key if they trust the issuing CA. If a subscriber trusts a particular CA to correctly associate identities and public keys, then the subscriber can load that authority's public key into his or her cryptographic application. Any public key certificate whose signature can be verified with the public key from the "trusted" CA certificate list (trust list) and is not listed on the CRL is considered valid. This means that the subscriber's public key can be extracted from that certificate with confidence that it really belongs to the subscriber.

Many CAs are required to support the validation process. Figures 8.1-3 and 8.1-4 illustrate several approaches for relying parties to address the problem of validating their certificates issued by the numerous CAs in use.

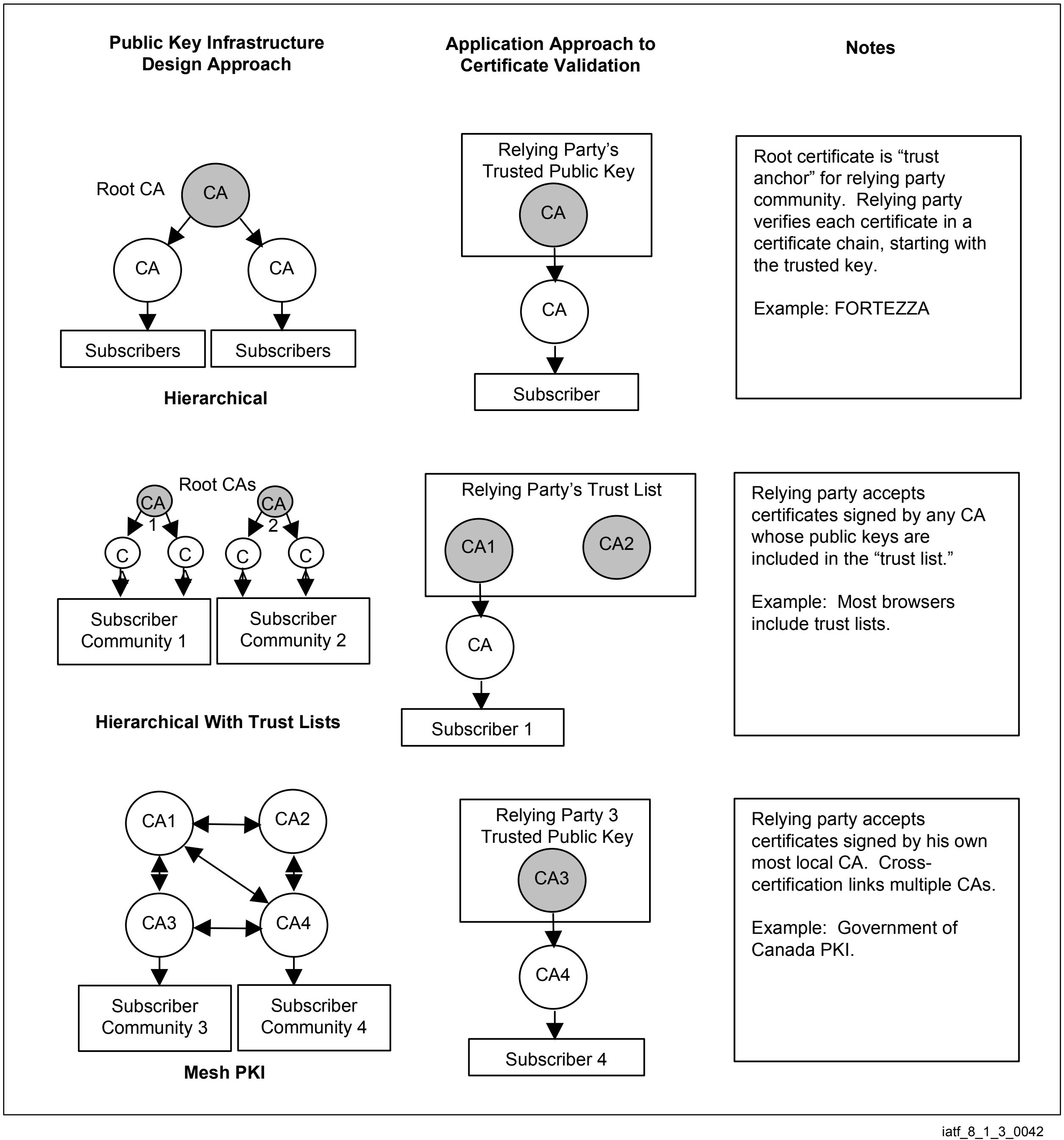

Figure 8.1-3. Hierarchical, Trust List, and Mesh Approaches to PKI Interoperation

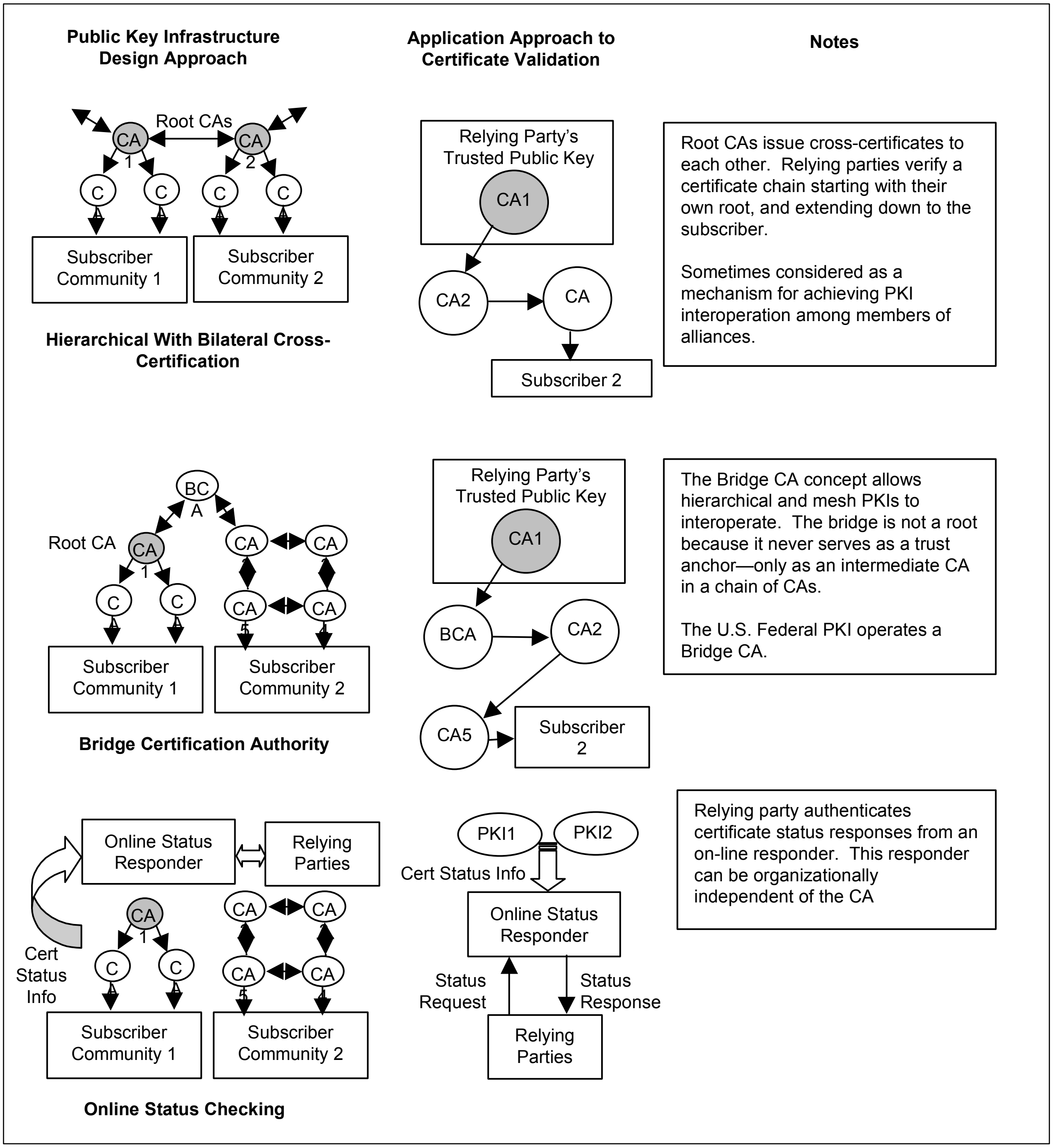

Figure 8.1-4. Bilateral Cross-Certification, Bridge CA, and Online Status Approaches to PKI Interoperation

Large PKIs will often support many CAs. The CA may “certify” the public key certificates of other CAs. When CAs do this, they are stating that certificates issued by the certified CAs should be trusted. PKIs are often composed of a hierarchical arrangement of CAs, with a Root CA at the top of the hierarchy. In this way, many CAs may be certified on the basis of approval by the Root CA that serves as a “trust anchor” for the PKI relying parties.

Although hierarchical PKIs have proven very popular for hierarchical organizations, many relationships within or among organizations are not hierarchical, and hence hierarchical arrangements of CAs are not always practical. For example, relying parties in the Federal Government will sometimes wish to authenticate public keys that originated from the commercial entities, academia, and foreign partners—none of whom will tolerate a subordinate hierarchical arrangement of CAs with the Federal Government.

A common way to deal with the problem of multiple nonhierarchical PKIs is to load multiple CA certificates into the verifying applications to be used as *trust anchors.* Most commercial Web browsers already have over 50 "trusted certificates" preloaded in their trust lists by the Web browser vendors. Subscribers may add other trusted certificates to this list, or delete the ones that are already there. So long as the certificate being verified was signed by a CA whose certificate is loaded in the trust list, or has a chain of certificates that terminates in a certificate included in the trust list, then the verifier considers the signer's certificate to be valid.

Another approach to dealing with nonhierarchical PKIs is bilateral cross-certification, which does not require a superior-inferior relationship between the CAs as is the case in hierarchical CA PKIs. Rather, two CAs—wishing to establish mutual trust among their two subscriber communities—issue certificates to each other that certify each other's public keys. PKIs that implement such bilateral cross-certification schemes are sometimes called mesh PKIs, to distinguish them from hierarchical PKIs. Hierarchical and mesh PKI schemes can be combined. For example, it is possible for the Root CAs for two hierarchical PKIs to cross-certify on a peer basis.

A special case of the mesh PKI is the Bridge CA (BCA). A Bridge CA issues cross-certificates to Principal CAs for multiple PKIs, thus reducing the burden of bilateral cross-certification. The Federal Government is deploying a BCA, which is expected to be the primary mechanism for cross-Federal PKI (FPKI) interoperation. The Federal BCA is discussed further in Section 8.1.7.4, U.S. Federal Public Key Infrastructure.

In either hierarchical or mesh PKIs, the signature verifier must build a chain of certificates that extends from the signer's public key to the CA that the signature verifier trusts. The verifier then must verify the signatures and check the revocation status for each certificate in the resulting chain. If each certificate in the chain is valid, then the verifier may consider the signer's public key to be valid.

An approach to certificate validation that breaks with the entire notion of certificate chains is that of online certificate validation. Online certificate validation involves sending a certificate to a networked resource that has been programmed to accept or reject certificates based on the organization's validation criteria.

Each approach to achieving interoperation among multiple PKIs has advantages and disadvantages, and each has aspects that must be considered carefully if security is not to be degraded as the community of interoperation is expanded. A full discussion of these factors is beyond the scope of this document, but discussed below are a few important points for each of the more common approaches to achieving cross-PKI interoperation.

8.1.2.1.1 Hierarchical CAs

Advantages:

- Many applications process hierarchical PKI certificates well.
- Relatively straightforward means for a large organization to enforce an organization certificate policy on a large community by revoking certificates from "subordinate" CAs not complying with the Certificate Policy.
- Application certificate processing is relatively straightforward.
- Only one "Root CA" certificate needs to be distributed to the applications via "out-of-band" authenticated channels to provide trust in a large number of certificates issued by subordinate CAs.
- Revocation of the subordinate CAs in the hierarchy is straightforward.
- Strong mitigation of "transitive trust" concerns exist; all trust decisions are made within the hierarchies of trusted PKIs (see disadvantages under "mesh PKIs").
- Large subscriber community can be managed using a relatively few CA certificates, providing ease of management.
- Hierarchical PKIs are usually interoperable with applications implementing trust lists.

Disadvantages:

- If the Root CA certificate is compromised, the hierarchical CA's entire subscriber population is at risk, and all subscribers must load new root certificates. Consequently, Root CA keys are normally very carefully protected.
- Hierarchical arrangements of PKIs often do not parallel organizational relationships; nonhierarchical organizations (e.g., collections of allies) often reject a hierarchical arrangement of CAs.
- PKI components based on an assumption of applications requiring only hierarchical CA elements may not be able to cross-certify, and such elements may not be able to interoperate with applications implementing mesh PKIs.

8.1.2.1.2 Trust Lists

Advantages:

- Commonly available in commercial applications.
- Relatively simple application certificate processing software.

- Provides a mechanism to provide a "per-CA" trust/do not trust decision for each instance of deployment of a public key using application.
- No centralized management required.
- Very flexible.
- Compatible with other mechanisms of achieving trust; use of trust lists in one PKI domain does not preclude interoperation with other PKIs using other mechanisms.
- Compatible with hierarchical PKIs.
- Strong mitigation of "transitive trust" concerns—all trust decisions are made locally, or within the hierarchies of trusted PKIs (see disadvantages under "mesh PKIs)."

Disadvantages:

- Management of the trust list often depends on local network administrators—or even individual relying parties—who often either do not understand PKI technology or do not have a basis for making informed decisions regarding which CAs should be trusted and which should not.
- Many applications are preloaded with dozens of CA certificates. Relying parties often accept all certificates issued by these CAs, without knowing anything about the level of assurance provided by the certificates these CAs issue.
- Modification of the trust list must be made relatively simple and hence may be relatively easy to subvert (technically or via faulty procedures).
- There is no straightforward revocation mechanism. If an organization wishes to stop trusting another CA, then the word must be spread to the organization's relying party population, and each network administrator or individual must remove the revoked CA from all applications manually.
- PKI elements based on assumptions of trust lists may not be able to cross-certify, and applications that rely on cross-certification cannot interoperate with such PKI components.

Note that some applications allow authenticated distribution of centrally managed trust lists, which mitigate (in some cases, eliminate) many of these concerns.

8.1.2.1.3 Mesh PKIs

Advantages:

- Allows CA trust relationships to mirror business or other nonhierarchical trust relationships.

- Relieves individual subscribers and their network administrator of the burden of maintaining trust lists.
- Not susceptible to the security vulnerabilities associated with distributed management of trust lists.
- Compromise of any CA certificate affects only the subscribers of that CA; there is no "Root CA" certificate whose compromise would be catastrophic.
- Applications designed to validate mesh PKIs also can usually validate hierarchical PKIs if a cross-certification exists between the mesh and the hierarchy.

Disadvantages:

- Developing and verifying chains of certificates from large mesh PKIs requires complex application software, and can have negative performance impacts.
- Because CAs are certifying other CAs, which may certify yet other CAs in other organizations, the arrangement of the mesh structure and the certificate security extensions must be very carefully managed to prevent the certificate chains from reflecting unintended trust relationships. This issue is sometimes called the "transitive trust" problem.
- Applications based on trust lists or hierarchical PKI concepts cannot interoperate with mesh PKIs without modification.

8.1.2.1.4 Online Certificate Validation

Advantages:

- Is simple application software.
- Relieves relying parties of the need to manage trust lists.
- Avoids the security vulnerabilities of managing trust lists.
- Avoids the management difficulties associated with mitigating transitive trust for mesh PKIs.
- Allows very rapid dissemination of revocation data. Note that most other methods (e.g., trust lists, hierarchical and mesh PKIs) use CRLs, which applications pull from directory systems for revocation notification. These CRL-based revocation notification methods can be as rapid as online checking, depending on the frequency of CRL updates and the details of the directory implementation. Online status checking can be seen simply as use of a special protocol for accessing a centralized trust/revocation list. The speed of revocation for such online methods depends on how often the centralized trust list/revocation list is updated, rather than on the speed of the online validation transaction. Conversely, for large PKIs with distributed directory systems, CRL

distribution and hence revocation notification can be slowed as a result of directory replication schemes.

- Allows applications to be compatible with all other PKI concepts because the online responder can implement virtually any certificate verification technology.

Disadvantages:

- Requires reliable network connections between the online validation responder(s) and all relying parties; relying parties not able to access the responder cannot process certificates.
- Believed by some analysts that the centralized nature of online responders creates scalability problems, though such responders can be "mirrored" or replicated (perhaps at the cost of introducing the performance delays associated with directory replication).

Note that regardless of the approach to how PKIs provide for cross-domain interoperation, relying parties can establish trust only in certificates they can obtain. Many application protocols provide some or all of the signature certificates and CA certificates necessary to verify subscriber signatures, but for public key applications that encrypt data, the relying party must obtain the subscriber's encryption certificate before encrypting the data. This transfer of the encryption certificate (sometimes called a key management certificate or confidentiality certificate) can be accomplished via an "introductory" message between the subscriber and the relying party, or the relying party can obtain the certificate from a certificate repository—often a directory system. See Section 8.1.4, Infrastructure Directory Services.

8.1.2.2 Security Services

The PKI plays a pivotal role in the generation, distribution, and management of the keys and certificates needed to support the public key-based security services of authentication, integrity, nonrepudiation, and confidentiality. The PKI itself employs some of the security services of confidentiality and integrity. Encryption is applied to private key material that is generated, stored, and distributed by the PKI to keep the private keys confidential. Integrity services are provided to the public key material that is certified by the PKI. The digital signature on a public key certificate binds a subscriber's identity with the public key, ensuring that the integrity of the public key contained within the certificate is maintained.

8.1.2.3 Infrastructure Processes

A variety of processes or functions are associated with the operation of a PKI that will be described in this section. This section is organized to reflect the KMI/PKI process categories that were described in Section 8.1.1.3, Infrastructure Process, and summarized in Table 8.1-3. Not all of the KMI/PKI process categories apply to certificate management; processes that do not apply will be indicated.

The type of applications that PKI is supporting affects certain PKI processes. This section describes the processes in the context of two public-key based applications: secure Web and secure messaging. These applications were selected because of their pervasive nature and because they illustrate the differences between real-time (secure Web) and store-and-forward (secure messaging) applications. Within this section, the differences in PKI processes that result from the influence of these different applications will be indicated. Key and certificate management for Web browsers and servers is described to show the PKI support required to enable secure Web communication via the Secure Sockets Layer (SSL) protocol. Key and certificate management associated with e-mail clients is described to show the PKI support required to enable secure messaging via secure messaging protocols such as Secure/Multipurpose Internet Mail Extension (S/MIME).

After reading this section, one will note that the majority of certificate management processes are transparent to subscribers of the PKI. Subscribers only need to know the name of the CA and either its e-mail address or Universal Resource Locator (URL) to communicate with it. Subscriber action is required in order to generate key material and obtain certificates used within secure applications such as web and messaging. However, this interaction is part of the security configuration for a secure product and is usually accompanied by an instructive subscriber interface.

8.1.2.3.1 Certificate Policy Creation

A Certificate Policy states the following:

- The community that is to use a set of certificates.
- The applicability of those certificates (that is, the purposes for which the certificates are appropriate).
- The common security rules that provide relying parties with a level of assurance appropriate to the community and their applications.

Before a PKI issues certificates, it should define its Certificate Policy and provide mechanisms to ensure that policy is being enforced by the PKI elements. In fact, one can consider a PKI to be nothing more than an organization's approach to generating and managing certificates in accordance with its certificate policy. This topic is discussed further in Section 8.1.5.1, Policy Creation and Management.

8.1.2.3.2 Registration

The registration function is defined in Section 8.1.1.3, Infrastructure Process, as the "authorization of people to make decisions about the validity of the subscriber actions." In general, the person responsible for decision making in the PKI context is a Certificate Management Authority (CMA). CMAs may be CAs (if they sign certificates or if they are responsible for a facility that automatically signs certificates) or an RA, if they simply provide the CA or CA facility with registration information. In any case, the CMA is responsible for reviewing certificate requests and verifying the information contained within the requests before generating certificates. The CMA operator is also responsible for authenticating the identity of the certificate requester to ensure that the proper identity is bound to the public key contained in the certificate.

When an organization establishes a PKI, it will identify the personnel who will be the CA and RA operators. The qualifications (e.g., clearances, training) for the personnel who assume these roles are often outlined in the PKI's Certificate Policy (or sometimes in a Certification Practices Statement [CPS]). The CA and RA operators must also be registered with the CA or RA software being used within the system. These operators normally have special accounts that will gain them access to the administrative functions performed by the CA or RA component. To access these accounts, the operators will need to authenticate themselves to the CA or RA components. Forms of authentication include the use of passwords, public key certificates, or hardware tokens and will depend on the capabilities of the CA or RA components used within the PKI.

Although most security products in use today require subscriber intervention in the key generation and certificate request process, other models need to be considered. One model is the case in which an organization requests a set of certificates on behalf of its subscribers. In this case, the organizational representative who submits the list of subscribers requiring certificates to the PKI may need to be registered with the PKI before submitting the list. Registration will assure the PKI operators that the organizational request is submitted from an authorized source. Many CA products available today have or are adding preauthorization features that will allow them to support this organizational registration model. Subscriber intervention is still required in the actual certificate request and response process to ensure that the proper key material and certificates are installed at the subscriber workstation.

8.1.2.3.3 Ordering

The primary function associated with ordering in a PKI is the request for a certificate. Certain PKIs may also generate key material for a subscriber. In these PKIs, the request for a certificate will also result in the generation of a public/private key pair. Discussions of key generation by the PKI will be described in subsequent releases of the Framework in Section 8.1.2.3, Infrastructure Processes. The remainder of this discussion assumes that the subscriber generates the public/private key pair and is indicative of the majority of secure applications in use today.

The certificate request process for Web browsers is described in detail. Differences between this process and the certificate request processes for Web servers and for S/MIME electronic mail clients will then be described briefly.

Web Browser

Figure 8.1-5 shows the first set of steps involved in obtaining a client certificate that is installed in a Web browser. The focus of these steps is on key generation and certificate request generation. The subscriber begins the key generation and certificate request process by directing the Web browser to connect with the CA Web front-end. The subscriber then fills in the certificate request HyperText Markup Language (HTML) form that is presented by the Web front-end. After completing the form, the subscriber presses the submit button on the form. An HTML tag (KeyGen) that appears on the form triggers the browser to generate a key pair for the subscriber. If this is the first time a subscriber has generated key material using the browser, the subscriber will be prompted to provide a pass-phrase. This passphrase is used to encrypt the subscriber's key material when it is stored in the key database that is located either on a floppy diskette or on the subscriber's workstation. When the subscriber needs to use the key material, the subscriber will be required to supply the pass-phrase, so that the material may be decrypted.

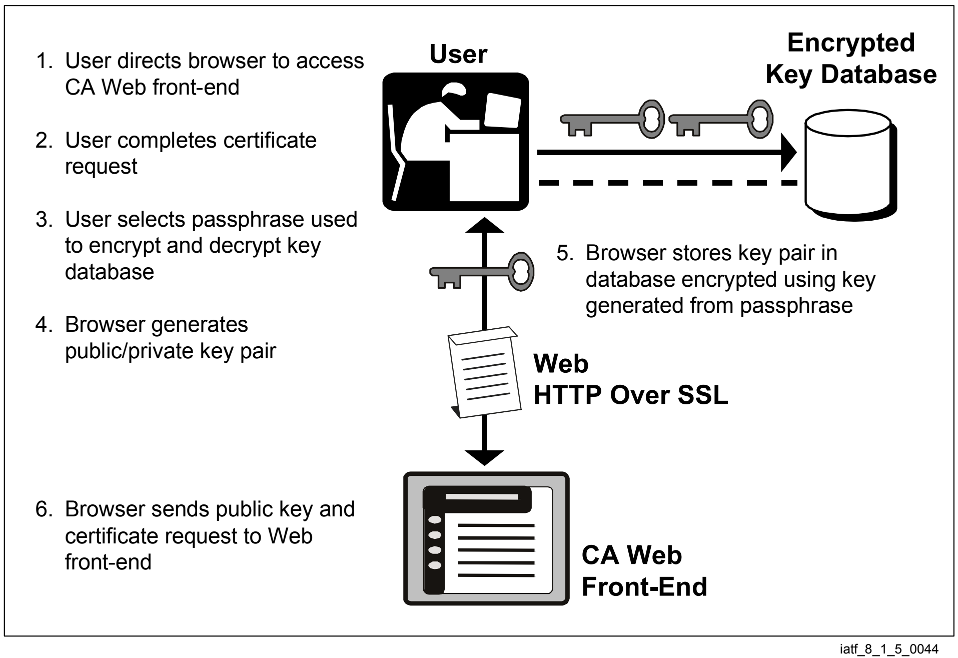

Figure 8.1-5. Browser Certification: Key Generation and Certificate Request

After the key material is generated, the browser provides the certificate request, which includes the public key and the information from the completed HTML form, to the server via a HyperText Transfer Protocol (HTTP) "PUT." Most Web browsers available today support either the Public Key Cryptography Standard (PKCS) 10 [2] or Netscape proprietary certificate request format. Both formats are self-signed, which means that the private key corresponding to the public key contained within the request is used to digitally sign the request. The CA verifies the digital signature on the request before generating the certificate. This verification ensures that a private key associated with public key being certified exists and that the certificate request had not been modified in transit. Obviously, the self-signed certificate request can be spoofed. If the certificate request is captured in transit, the public key and corresponding certificate can be replaced. To counter such a threat, CAs usually only accept certificate requests across a secure channel such as an SSL-encrypted session between the browser and the CA Web front-end.

The CA stores the certificate request until the RA or CA operator approves it. Some CAs provide a reference number to the subscriber, which the subscriber can use to make inquires regarding the status of the certificate request or to download the completed certificate.

Figure 8.1-6 shows the steps conducted by the CA to process the certificate request received from the subscriber. The certificate request approval and certificate generation process—depicted in Figure 8.1-6 and described below—assumes that the CA provides a RA function. Noted that the architectures of CA products vary. Not all CAs have a RA component, nor can they be configured to provide such a function. If there were no RA function available, then the CA operator would conduct all steps within the certification process.

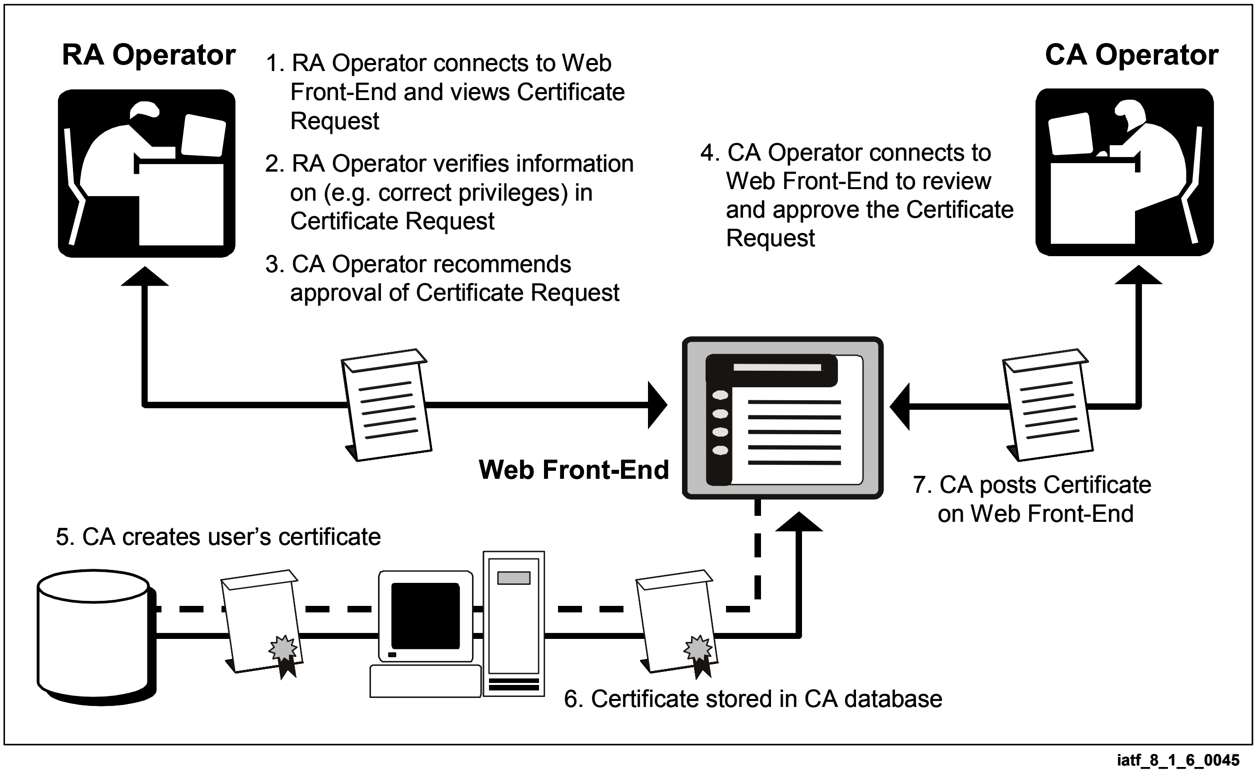

Figure 8.1-6. Browser Certification: CA Processing Request

The RA accesses the Web front-end to review any pending certificate requests. The RA displays the information contained in the request and verifies that it meets the policies set by the CA (e.g., if the subscriber's Distinguished Name [DN] follows the proper format or if the subscriber's key is of a certain length). If further information is required before the request can be processed, the RA can contact the subscriber who submitted the request. Other procedural activities, such as requiring the subscriber to be authenticated in person by the RA, may also be implemented at this point.

Web Server

The procedure for generating a certificate for a secure Web server is similar to generating a subscriber certificate for installation into a Web browser. Most secure servers provide a forms-based interface for the Web server administrator. One of the options available through the form is to generate and install server certificates. The administrator performs the following steps for generating and installing the Web server's certificate.

The first step is to run the key generation program at either the command line or via a graphical user interface (GUI). The steps for generating the public and private key pair are similar to generating a subscriber's public and private key file. The administrator must specify a file name to which the new key pair file will be stored. The administrator may need to generate random information to initialize the random number generator. Finally, the administrator must supply a passphrase that will be used to protect the key pair. After the administrator has created the server's key pair file, the administrator fills out the server's certificate request. The certificate request contains information including the server's DN and the administrator's e-mail address and telephone number. Web servers use the PKCS 10 certificate format. After the form is completed, the administrator can send the form to the CA. E-mail is the transport mechanism presently used by Web servers to submit certificate requests and receive certificates.

The CA process for a server certificate request is essentially the same as that of the Web browser. The only difference is the request is received at the CA via e-mail and the certificate is returned to the server via e-mail. In-person authentication of the Web server is also not feasible. The CA operator can confirm information about the server request with the system administrator by requiring that the administrator appear in-person at the CA or requiring that documentation be provided by the server's owning organization, which states that the server is located at that organization and requires a certificate.

S/MIME Client Certification Process

Figure 8.1-7 shows the certification process for a generic S/MIME client. Using the security configuration options of the S/MIME client, a key pair for the subscriber is generated locally. The private key is stored in the key's database of the product. This database is protected by a key computed from hash of a pass-phrase provided by the subscriber at key generation. The public key is placed either in a self-signed certificate or in a certificate request. This description focuses on the latter option, which requires interaction with PKI components. S/MIME clients support the PKCS 10 certificate requests, which are transported to the CA via e-mail (Simple Mail Transfer Protocol [SMTP]) using a *smime.p10* message format.

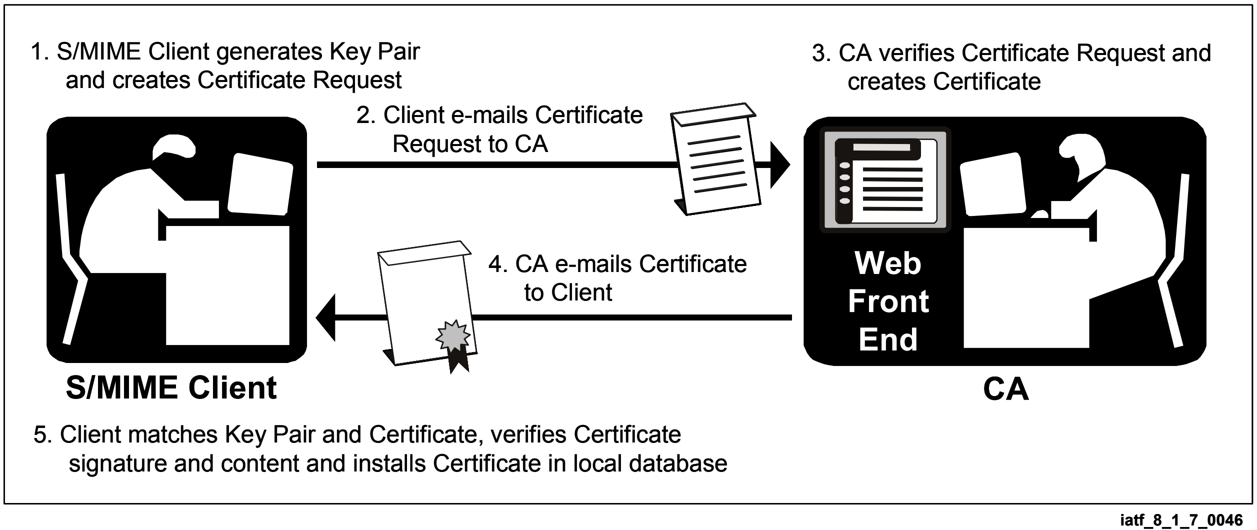

Figure 8.1-7. S/MIME Client Certification Process

The certificate request is received at the CA via e-mail. Once the request is received, the actual generation process for the S/MIME client certificate is essentially the same as that which was followed for the Web browsers and servers previously described. In-person authentication of the subscriber may be implemented by the CA if so desired.

8.1.2.3.4 Generation

In the context of PKIs, there are two aspects of generation: key generation and certificate generation. Both generation aspects are described in this section.

Key Generation

In public key management, the generation of key material is closely tied with the request for a certificate. Therefore, Section 8.1.2.3.4 is written following a distributed model where the key material is generated locally in the context of the secure application. It is also possible to generate public key material following a centralized model where the CA or some other trusted entity would generate the key material on behalf of the subscriber or application. Because keys are generated in a single place and using only one system, centralized key generation offers the opportunity to use better equipment (e.g., cryptographic hardware, random number generators, and techniques within the key generation process). Centralized key generation is often used in environments with very strong security requirements. In addition to the location of the key generation, the models also differ in the type of additional key management functions that are required to support each model. When the key material is generated locally, the private key stays within the control of the subscriber or application from its generation to its destruction. Only the public key needs to be conveyed to the CA for inclusion in the certificate that the CA will subsequently distribute to the subscriber or application. When the key material is generated centrally, not only does the CA have to generate and distribute the certificate, but also there is the added function of securely distributing the private key to the subscriber or application.

Today, secure private key distribution is achieved through manual distribution or distribution via a secure protocol, which may be proprietary, specific to a product line, or a more widely accepted security protocol such as SSL.

Another consideration when generating key material centrally is how the key material is to be used. Usually only asymmetric key material that will be used for key or data encryption is generated centrally. Key material, which will be used for digital signature purposes, is normally generated locally. This is the preferred approach because one would like to use digital signatures to provide the security services of nonrepudiation. True nonrepudiation services can be provided only if the entity generating the signature key material is the only one who knows the private key. If the digital signature key material were generated centrally, then this would not be the case. In light of these considerations, asymmetric cryptographic products are now migrating to two key systems in which separate key material is used for data/key encryption and digital signature purposes. Commercial products are available that combine both the distributed and centralized key generation methods. These products generate key material associated with key or data encryption centrally and key material associated with digital signature locally.

Another topic associated with key generation is whether the key material is generated in software or in hardware. Many of the commercial security products available today perform all cryptographic functions, including key generation in software. However, concerns exist that software cryptography may not be adequate for all situations. Therefore, there has been a move to provide flexibility within security products to allow key material to be generated and cryptographic functions to be performed on hardware tokens, including both personal computer (PC) cards (a.k.a., Personal Computer Memory Card International Association [PCMCIA] Cards) and International Standards Organization (ISO) 7816 compliant smart cards. Note that many of the commercial CA products available today use hardware tokens or other types of cryptographic hardware to generate the CA key material and perform the cryptographic functions associated with the CA functions. When hardware tokens are used, there are added management functions associated with the tokens themselves, including their initialization, personalization for a particular subscriber, and distribution of the token and any personal identification number (PIN) associated with the token. Today, many of the token management functions are handled outside the context of the PKI. The FORTEZZA Certificate Management Infrastructure (CMI) is one notable exception. However, there appears to be a trend within the PKI arena to add token management functions to the growing list of functions provided by the PKI.

Another consideration associated with key generation is the length of the key material. In general, the longer the key length the stronger the key because it is more difficult to break longer keys. In the commercial cryptographic implementations in use today, asymmetric key materials are usually 1,024 bits long, with 2,048 bit or longer keys being used for more sensitive applications such as CA signing keys. Today, strong symmetric key implementations use 128 bit keys. Type 1 cryptographic implementations used to protect classified information use even longer keys. Note that export and import controls imposed by governments may restrict the key lengths within exportable or importable versions of cryptographic-based products.

Certificate Generation

As was done in the Ordering section, a full description of the Web browser certificate generation process is provided. Differences between this process and that of the Web browser and the S/MIME e-mail client are summarized.

Web Browser

Figure 8.1-8 shows the steps conducted by the CA to process the certificate request received from the subscriber. If all the information within the request is satisfactory and the subscriber is authenticated to the RA's satisfaction, the RA marks the certificate for approval. Depending on the configuration of the CA product, the certificate may be automatically generated once the RA has approved the request or CA operator intervention may be required to generate the certificate.

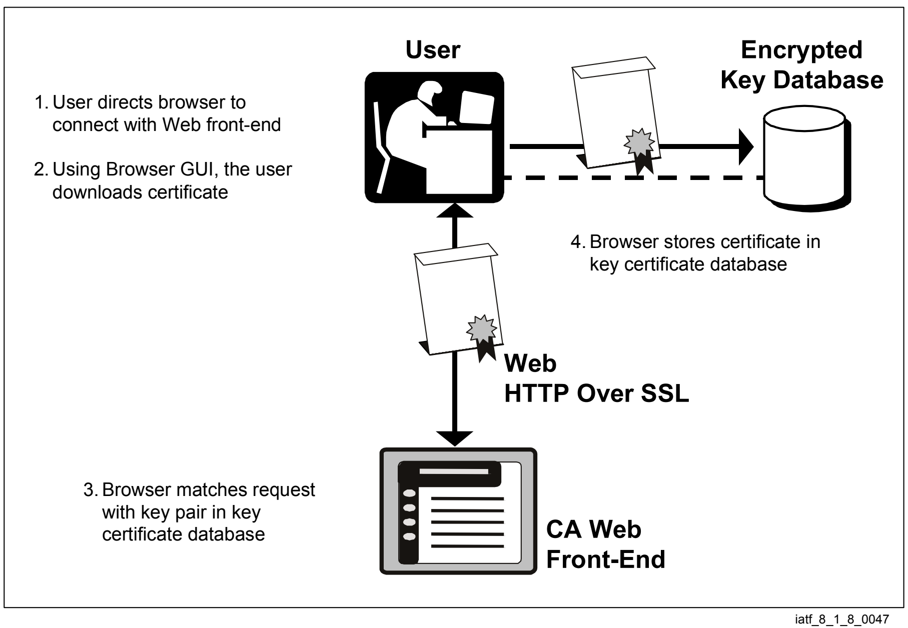

Figure 8.1-8. Browser Certification: Installing Certificate in Browser

Once the certificate has been created, a copy of the signed subscriber certificate is stored in the CA database and posted to the Web front-end. The subscriber can then download and subsequently use the certificate. Many CA products send e-mail to the subscriber to notify them that the certificate has been created and to provide them the URL where they may download the certificate. If the CA does not provide notification services, then the subscriber would need to periodically check the Web front-end to determine if the certificate is ready.

Web Server

The CA process for generating a server certificate is about the same as that of the Web browser. The only difference is that the certificate is returned to the server via e-mail.

S/MIME Client

The certification process for a S/MIME client was shown in Figure 8.1-7. The certificate generation process for the S/MIME client certificate is about the same as that which was followed for the Web browsers previously described. Once the certificate request is validated, the CA generates a certificate for the S/MIME subscriber. S/MIME clients expect to receive certificates back, in PKCS 7 [3] format via e-mail.

8.1.2.3.5 Distribution

Certificates can be distributed in several ways. The certificates can be e-mailed to the requesters, or the requester can download a copy of the certificate from the Web front-end of the CA or from a certificate repository such as a directory. This section describes the distribution options for certificates in the context of secure Web and messaging applications. As was done in the Ordering and Generation sections, a full description of the Web browser certificate generation process is provided. The difference between this process and that of the Web browser and the S/MIME e-mail client are summarized.

Web Browser

Once the certificate has been posted to the Web front-end, the subscriber can then download and subsequently use the certificate. Many CA products send e-mail to the subscribers to notify them that the certificate has been created and to provide them the URL where they may download the certificate. If the CA does not provide notification services, then the subscriber would need to periodically check the Web front-end to determine if the certificate is ready. Figure 8.1-8 shows the final set of steps that complete the certification process.

To download the certificate, the subscriber needs to direct the browser to connect to the Web front-end. The subscriber may supply a reference number supplied during the certificate request process to find their certificate that appears as a hotlink. The subscriber clicks on this link to start the download process. Following the set of subscriber screens that the browser displays, the subscriber accepts the certificate for download. The certificate is downloaded and stored in the keys database where it may subsequently be referenced. As part of the download process, the browser software checks that the private key associated with the public key contained in the certificate is located in the key database. If the associated key is not found in the database, the software will not download the certificate and will provide an error message to the subscriber.

At certificate retrieval time, the subscriber may also need to download certificates associated with the CAs within its certification path. Usually the CA certificates are also available for download via a Web interface. The download of the CA certificate is about the same as that of a subscriber certificate. The browser stores the CA certificate within its certificate database. However, the browser does differentiate between subscriber and CA certificates and considers the CA certificates to be trusted, meaning that certificate path validation will terminate once a CA certificate in the path is not found in the certificate database. The browser is able to identify CA certificates from subscriber certificates, because different HTML tags are applied to each type of certificates. Note that Web browsers are distributed with a number of well known root CA certificates (a trust list) already installed in the certificate database. These certificates are usually associated with vendors that provide certification services. It is possible to modify the certificate database and delete any CA certificates that one does not want to be trusted within a specific environment.

Web Server

Web server certificates are distributed to the server via an e-mail message from the CA. Certificates are often sent in a PKCS 7 [3] SignedData formatted message. This message format allows the full certification path (server and CA certificates) associated with the subscriber to be conveyed in the same message. Once the administrator receives the certificate from the CA, the administrator can install the certificate into the server. Most servers provide a GUI for this step. The GUI typically asks for the pathname to the file containing the certificate, or the certificate can be pasted into a text block on an HTML form. The Web server will then automatically install the certificate in the Web server's encrypted key database. As part of the download process, the server software checks that the private key associated with the public key contained in the certificate is located in the key database. If the associated key is not found in the database, the software will not download the certificate and will provide an error message to the administrator. Any CA certificates found in the PKCS 7 message will be installed within the certificate database of the Web server. Like Web browsers, the CA certificates are considered trusted and are indicated as such in the certificate database of the Web server.

S/MIME Client

An S/MIME client receives the certificate back from the CA in an e-mail message. Like Web servers, S/MIME certificates are sent in a PKCS 7 [3] SignedData formatted message. Once received at the client, the message is opened by the subscriber. The S/MIME client provides functionality that verifies the PKCS 7 formatted message and automatically installs the client certificate and any CA certificates in the local certificate database. As with both the Web browser and server, the S/MIME client also differentiates CA certificates from subscriber certificates within its database and is normally distributed with popular root CA certificates installed. However, unlike the Web products, most S/MIME products do not automatically trust CA certificates installed in the client. Normally, the subscriber will need to explicitly mark the certificate as trusted before the client will recognize the certificate as trusted.

8.1.2.3.6 Compromise Recovery

This section describes how the PKI notifies its subscribers when certificates are revoked and assists its subscribers in recovering from a compromise of key material. Recovery of the PKI itself from a compromise will be described in Section 8.1.5.8, Compromise Recovery.

There will be instances when the certificates issued by a CA need to be revoked. Revocations fall into two major categories: security compromise revocation and routine revocation. Security compromise revocation covers instances when the associated private key material has been compromised, when a subscriber no longer can gain access to the private key (e.g., forgotten PIN or password or lost token), or if the subscriber has been fired or stripped of privileges granted by an organization. Report of such compromise should be immediate, and the actual revocation of the certificate by the CA should occur immediately. Routine revocation covers cases in which certificates need to be revoked because information contained within the certificate is no longer valid for a variety of reasons (e.g., name changes [marriage/divorce]) or a change of organizational affiliation. These types of revocations also need to be reported to the CA.

Regardless of the reason, for compromise it is important that the CA be notified about the need for revocation. Thus, a certificate revocation notice is sent to the CA that issued the certificate. This certificate revocation notice may take many forms, including an e-mail message, a phone call to the CA operator, the submission of some other type of form, or some combination of the above. It is important that the CA operator ensure that the revocation notice is authentic before revoking a certificate to prevent denial of service attacks. The request may be authenticated in various ways, including the use of a digitally signed revocation notice, the provision of a password, or in-person authentication. Commercially available CA products are only beginning to add automated certificate revocation notification to their products; therefore, the variety of authentication options is likely to grow.

A CA notifies other subscribers when a certificate has been revoked through the issuance of CRLs. A CRL contains certificates still within their validity interval, but that no longer represent a valid binding between a public key and a DN or privilege. Certificates must remain on the CRL until their expiration date. The CA will periodically generate and distribute CRLs. CRL distribution mechanisms are usually the same as those employed for certificates; CRLs are posted to directories, made available via a Web interface or distributed via e-mail.

The distribution and process associated with a CRL is one of the major issues faced within the PKI community today. There is a concern about the timeliness of revocation notification because CRLs may only be generated periodically. To counter this issue, emergency CRLs or CRLs containing only certificates revoked because of compromise may be distributed on a more frequent basis and may be pushed to the subscribers versus just posted to a repository where the subscriber may need to go to retrieve the CRL. Another major concern is that the CRLs may grow rather large, especially as the number of certificates issued by a specific CA increases. The size of a CRL will affect the time it takes to validate a certificate path. Finally, there is often no consolidated directory from which applications can obtain CRLs. Because of these problems, many of the security products available today do not provide an ability to process CRLs, or the subscribers must resort to manual methods to remove a revoked certificate from the databases of these products. At the same time, there is ongoing research exploring alternative certificate revocation models.

One such alternative is the online validation of a certificate. In this case, a certificate or certificate path may be sent to a trusted entity—which may be a CA or a certificate repository—which will determine if the certificate(s) is valid and notify the requester of the results. Online validation also brings its own set of concerns. Online validation requires that there be network connectivity between the requestor and the trusted entity performing the validation. The availability of the network and the added network traffic resulting from the validation requests and responses are considerations associated with implementing online validation. The level of trust needed in the entity performing the validation is also an issue and will depend on the requirements of the environment in which one is operating.

A CA also assists subscribers in their recovery from a key compromise. In the case where the CA has been involved with the generation of the key material or the initialization of a token, the CA may offer backup functionality. In this case, if the subscriber has lost access to the key material and needs to recover information that may have been encrypted in that key material, the CA may be able to provide a copy of the key to the subscriber or issue a new token with the old key material provided. In the case of a security compromise, the subscriber will need to have a new key pair and certificate generated. The CA will be involved in this process to the extent it was involved in the initial key and certificate generation process that was described earlier in this section.

8.1.2.3.7 Accounting

A number of auditing functions associated with the PKI are described in Section 8.1.5.7, Accounting.

8.1.2.3.8 Key Recovery

A PKI may provide key recovery functionality by providing key backup or escrow of key material. Key backup or escrow capabilities are normally provided only to asymmetric key material that is used to encrypt either data or keys and not to key material used for digital signature purposes. Key backup or escrow capabilities can be provided when the CA generates the keys on behalf of the subscriber. In this instance, the CA will store a copy of the private key in a secure database. This key material may be retrieved from the database and used to recover information encrypted with the material if the need arises. It is possible for a CA to provide backup capabilities even when the subscriber generates the key, but this raises the issue of how the private key is securely sent to the CA for backup. It is also possible that a completely separate infrastructure other than the PKI can be used to support key recovery.

8.1.2.3.9 Rekey

During the course of PKI operations, it will become necessary to renew certificates. There are two cases for renewal: one is when the certificate reaches its natural expiration date, whereas another is when the previous certificate has been revoked and a new certificate needs to be issued. For the first type of renewal, there are two subcategories: a renewal where both a new key pair and certificate are generated, or a renewal where the key material is not changed but a new certificate is created. Whether a new key pair is generated is dependent upon the recommended key life span. If the key life span and certificate validity period coincide, then new key material should be generated at renewal. However, if the key life span is longer than the certificate validity period, then it may be possible to recertify the key material, until its recommended life span is reached.

Certificate renewal with rekey is about the same as the generation of an initial certificate, whereas the renewal without rekey may be a somewhat simpler process. CA products today vary in their renew capabilities and may limit the amount of information within the certificate that can be changed at renewal time.

8.1.2.3.10 Destruction

Unlike symmetric key management, the PKI is not normally involved in tracking the destruction of key material. When asymmetric key material reaches its expiration date or when it has been compromised, it may be destroyed. The subscriber would normally do the actual destruction of the material. At this time, most security products require that a subscriber manually remove old keys and certificates from the database. Note that there are instances when a subscriber would need to retain key material even after its expiration or compromise in order to be able to recover data encrypted in this key material. In this instance, the subscriber (or an agent acting on behalf of the subscriber) will need to retain the material until access to the encrypted data is no longer needed or when the encrypted data has been re-encrypted in new key material.

8.1.2.3.11 Administration

Administration functions for the PKI are described in Section 8.1.5.12, Administration.

8.1.2.4 Requirements

Overall security requirements for PKIs are specified in a Certificate Policy, which describes requirements imposed both on the operation of the PKI and on PKI subscribers. General requirements for a KMI/PKI that are common to many Certificate Policies are found in Section 8.1.1.4, Requirements. PKI subscriber requirements commonly found in Certificate Policies are described in this section. Requirements specific to the operation and maintenance of the PKI itself are described in Section 8.1.5, Infrastructure Management. Requirements related to the use of PKI services include the following:

- Subscriber generated asymmetric key material shall be generated securely.
- The subscriber shall protect the private key material from disclosure and shall also protect any password or PIN used to access the private key material.
- A subscriber shall provide accurate information to the CA when requesting a certificate. In other words, the subscriber shall provide the appropriate identifying information and the appropriate public key for certification.
- The subscriber shall only use the private key and associated public key certificate for applications or purposes approved by the PKI. Approved applications are normally documented in the CPS for the PKI.
- The subscriber shall notify the CA when the private key has been compromised, or if other information within the certificate becomes invalid.

- The subscriber shall obtain the public key of the Root CA and any CA public key certificates from an authorized source in a secure manner.
- If an organization requests certificates on behalf of a group of subscribers, the organization's representative shall provide the CA with an accurate list of subscribers to whom certificates shall be issued.

8.1.2.5 Attacks and Countermeasures

The strength of the security services provided by a cryptographic capability such as digital signature depends on a variety of factors, including the security of the underlying cryptographic keys, the strength of the binding between the subscriber identity and public key, and the specific application implementation. As a result of the PKI's role in the generation, distribution, and maintenance of private and public keys and certificates, threats to the PKI are of concern. If the PKI operates as expected, the confidentiality of private keys and the integrity of public keys should be maintained. However, it is possible that threats to the PKI—be they intentional or unintentional—may result in the disclosure of the private keys or in the modification of the public keys. Other threats to the PKI can lead to the denial of services provided by the system. This section focuses on the attacks and countermeasures specific to the PKI. These attacks and countermeasures are discussed from the perspective of the subscribers of a PKI. Infrastructure specific attacks and countermeasures are described in Section 8.1.5, Infrastructure Management. More general attacks and countermeasures for KMI/PKI can be found in Section 8.1.1.5, Attacks and Countermeasures.

8.1.2.5.1 Attacks

Attacks aimed at the PKI subscriber are designed to gain access to the subscriber key material, to modify or substitute the subscriber key material, or to deny the services of the PKI to the subscriber. Attacks include the following:

- **Sabotage.** The subscriber's workstation or hardware token on which key materials and certificates are stored may be subjected to a number of sabotage attacks, including vandalism, theft, hardware modification, and insertion of malicious code. Most of these attacks are designed to cause denial of service. However, attacks such as hardware modification and insertion of malicious code may be used to obtain copies of subscriber key material as they are generated or to obtain information entered by the subscriber such as a password.
- **Communications Disruption/Modification.** Communications between the subscribers and the PKI components could be disrupted by an attacker. This disruption could cause denial of service, but also could be used by the attacker to mount additional attacks such as the impersonation of a subscriber or the insertion of bogus information into the system.
- **Design and Implementation Flaws.** Flaws in the software or hardware on which the subscriber depends to generate and/or store key material and certificates can result in the malfunction of the software or hardware. These malfunctions may deny services to the

subscriber. The flaws may be accidentally or intentionally exploited to disclose or modify the subscriber's key material or certificates. Improper installation of the software or hardware may also result in similar consequences.

- **Subscriber Error.** Improper use of the software or hardware associated with the subscriber's interaction with the PKI or with the storage of keys and certificate generated by the PKI may also result in denial of service, or the disclosure or modification of subscriber key material and certificates.
- **Subscriber Impersonation.** It is possible that an attacker may impersonate a legitimate subscriber of the PKI. Depending on whether the PKI generates key material on behalf of a subscriber, the attacker may obtain both key materials and certificates in the name of the legitimate subscriber, or the attacker may substitute his or her own key material for that of the legitimate subscriber and obtain a certificate from the PKI.

8.1.2.5.2 Countermeasures

Countermeasures that can prevent or limit the attacks to subscribers of a PKI include the following:

- **Physical Protection.** Physical protection of the subscriber's workstation, communications link with the CA, and/or hardware tokens will counter many of the sabotage and communications disruption related attacks.
- **Good Design Practices.** Concerns over flaws in the software and/or hardware design may be alleviated if good design practices are followed during the development of the software and/or hardware used in conjunction with the PKI.
- **Testing.** Testing of the software and/or hardware may also be used to counter attacks to the system that result from the exploitation of flaws in the system.
- **Training.** Training of subscribers is vital to eliminating or at least reducing the possibility of inadvertent attacks due to subscriber error.
- **Strong Authentication.** Strong authentication of the subscriber by the PKI components greatly reduces the possibility of impersonation attacks.
- **Encryption.** Encryption of the link between the subscriber and the PKI components reduces the possibility that an attacker may eavesdrop on the communications and try to disrupt or modify the communications.
- **Contingency Planning/System Backup.** Backup of a subscriber's key materials, certificates, and relevant software and hardware is the best mechanism for protecting against design flaws that result in system failure.

A Certificate Policy describes all countermeasures a PKI requires to provide a level of assurance consistent with anticipated certificate usage.

8.1.3 Symmetric Key Management

8.1.3.1 Overview

Although overshadowed by PKI in the literature, Symmetric Key Management (SKM) remains an important technique in the real world. Most legacy systems use symmetric cryptography exclusively. Even with the expanding use of asymmetric techniques, many new and emerging applications, such as multicast, will still require secure symmetric key and asymmetric cryptography.

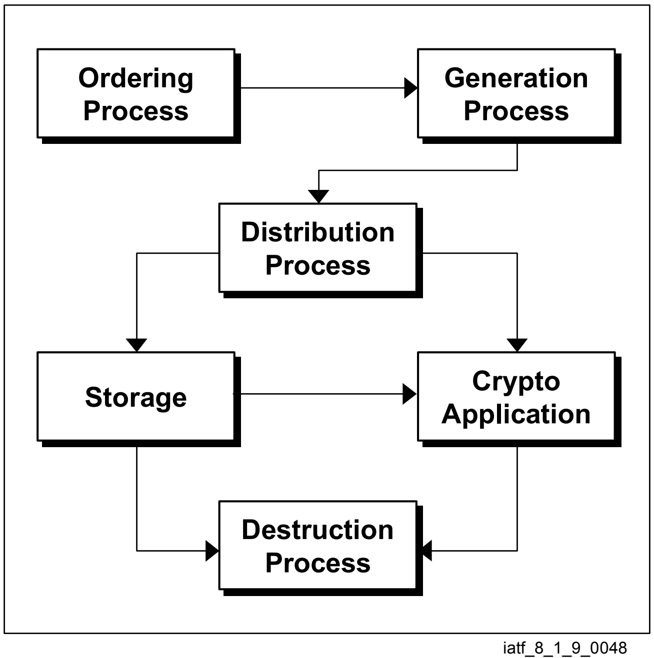

Figure 8.1-9. Critical Elements of Symmetric Key Management Activities

With a symmetric key algorithm, the encryption key can be calculated from the decryption key and vice versa. This is very different from the public key algorithm where it is presumed unfeasible to calculate the decryption key from the encryption key. In most of the symmetric systems, the encryption and decryption keys are the same, requiring the sender and the receiver to agree on a key before they can pass encrypted messages back and forth. Information on certificate based public-key algorithms can be found in Section 8.1.2, Certificate Management.

The old adage "good management is the key to success" could never be more true than in the application of symmetric key in the world of cryptography. The strongest of cryptographic algorithms are reduced to nil if the management of the keys used with the cryptography is poor. For symmetric key applications where a common secret key is required by all users, delivering the correct key to all the users and keeping them secret can be extremely complicated and expensive. Figure 8.1-9 depicts the critical elements of symmetric key management.

System requirements play heavily in the decision to use symmetrical key because there are significant advantages and disadvantages in its use. Many of the problems with SKM have become more complex as the community of cryptographic users has increased and become more geographically separated. Ordering, generation, distribution, loading key into cryptographic applications, storage, and key destruction are becoming more critical.

8.1.3.2 Advantages of Symmetric Key Technology

- Everyone in a communications network can use a single key for as long as necessary. The keys can be changed as often or as infrequently as the security policy allows.
- Local generation of keys can minimize many of the problems with ordering and distribution. There is no need to connect with a central authority.
- Key structures for symmetric key are extremely simple—predominately, a sequence of random numbers.
- Algorithms using symmetric key processing are usually much faster than their asymmetric counterparts. In many instances, asymmetric keys are used to securely distribute the symmetric keys to other users in the network.
- Symmetric keying supports netted and point-to-point operations.
- Symmetric keying limits who holds a specific key; therefore, no outside access control mechanisms are needed to control who talks to whom.
- Symmetric keys do not require extensive validation before use.
- Symmetric keys are not reliant on an extended trust path.
- Potentially fewer people need to be trusted in the ordering and distribution path.
- The creation of an unauthorized key is only dangerous when an attacker can get someone to use it in place of the correct key; consequently, alone it does no harm.

8.1.3.3 Problems with Symmetric Key

- One lost key will compromise the whole network, requiring the replacement of every user key.
- Limited cryptographic services (e.g., no nonrepudiation, implied authentication).
- There is difficulty scaling to large communities. There is an upper limit for the size of cryptographic networks using a common key.
- The larger the number of operators using a common symmetric key, the more likely the key will be compromised.
- Large amounts of symmetric key may need to be produced to meet potential compromise and contingency uses. This key must be securely delivered and locally stored.
- Distribution delay causes key to be generated and distributed well in advance of its use; allowing potential harmful access to the key for longer periods of time.
- Nets must be predetermined. It is difficult to create dynamic communication networks.

- Key must be kept secret at all times.
- Long cryptoperiods cannot be used for per-session communications.
- There is no intrinsic way to know who created the key.
- There is no back traffic protection. A compromise of a key at any time exposes all traffic encrypted using the key since the beginning of the cryptoperiod.

8.1.3.4 Critical Elements of Symmetric Key Management

Good key management with its many facets is vital for maintaining security. SKM involves the total life expectancy of a key; controlled processes should be established and maintained for ordering, generation, distribution, storage, accounting, and destruction of the key. There must be ways to detect compromised keys and provisions to resecure the system and efficiently determine the extent of any compromise.

- **Ordering.** Only authorized individuals should be allowed to order key and only keys for which they have been given explicit authorization to order. Because the symmetric networks must be predefined, the orderer must have access to the communication network management. They need to know what users will need the key and when they will need it. The key must be ordered so that it can be delivered to all users prior to them needing it. When the key is generated centrally, it may require ordering several months in advance of actual use, given the worldwide nature of many nets. The key management system must ensure that the orderer has the authorization to order the key as well as whether the recipient(s) are authorized to receive the key.
- **Generation.** Generation must be performed in a secure environment to prevent unauthorized access to the key. The best cryptographic algorithms can be nullified if the key falls into the wrong hands. The generation process must be able to produce the total set of acceptable keys for the specified encryption algorithm. Weak or sensitive keys associated with the specified algorithm must be deleted (e.g., DES has 16 weak keys). [4] Symmetric keys are usually random bit streams requiring a quality control process to ensure the randomness of the bit streams.
- **Distribution.** Symmetric key can be delivered in physical form depending on trusted people and technical protection techniques like tamper-resistant canisters. For very sensitive key, two-person control can be used to gain more assurance. These techniques, however, provide only minimal protection to the key over its life cycle. The more people having access to a key, the more likely it is to be compromised; therefore, a goal of secure distribution is to provide the key electronically directly from the generator to the user equipment through benign delivery techniques. Public key techniques can support benign delivery techniques. They allow the user equipment to create an authenticated session key with the generator to pass symmetric key. When true benign techniques are not possible (i.e., the user equipment does not have asymmetric cryptography), the key

should be protected in encrypted channels as long as possible. Electronic deliveries to an intermediate node close to the user may be a reasonable compromise.

- **Storage.** Keys must be stored when waiting for distribution to the user or when used as contingency key. Storage of unencrypted symmetric keys may be required to recover when a link goes down. The protection of these keys is critical. They must be stored securely. Physically distributed key can be protected only through strict physical and personnel security. Electronic keys should be stored in encrypted form where physical, personnel, and computer security mechanisms are in place to limit who can decrypt and access the keys.

- **Loading Key Into the Cryptographic Application.** Loading key requires a protected interface. Physical protection of the key at the interface is critical to prevent the key from being exposed where it could be copied or replaced. Although minimal protection is required for loading encrypted keys, a high level of protection is required for the less frequent loading of the corresponding protection decrypt key.

- **Destruction.** Many potential media exist with which symmetric key can be deployed. These media include paper (e.g., manual codebooks, key tape), mechanical components (e.g., plugs, boards), and electronic components (e.g., random access memory [RAM], electrically erasable programmable read only memory [EEPROM], programmable read only memory [PROM]). Because the compromise characteristics of symmetric key allow recovery of previously encrypted traffic, it is imperative that the keys not be stored any longer than necessary to perform their mission. At the end of a cryptoperiod, the secret key must be destroyed in all locations (including secondary sources like contingency storage and incidental electronic storage).

- **Compromise.** Symmetric keys are vulnerable to compromise (e.g., physical delivery, large cryptonets, long cryptoperiods), so compromise detection and recovery are critical. There are no technical mechanisms where the network can control the damage done through a compromise. The compromise of a secret key potentially exposes all the traffic it ever encrypted and invalidates the assumed authentication for future traffic. To recover from a compromise, each user must be notified and provided a new key. The major problems of this approach stem from the long time it might take to notify the users and then the length of time necessary to replace the keys. While users are being notified and taken off the net, other users may still be using the key thinking that it still protects the data. There are no technical mechanisms that can be used to ensure that all users have been notified. There is a significant denial of service issue bringing up a widely dispersed network. Even after a user has stopped encrypting on the compromised key, the user cannot communicate until the new key arrives, either from contingency stock or the generation of new key.

- **Accounting.** As a result of the distribution of keys to a large number of users potentially scattered around the world and the corresponding danger of a compromised key, additional mechanisms must be in place to track keys throughout their life cycle. Effective accounting improves the tracking of who had authorized access to a key, when and where key was delivered, and when a key was destroyed.

8.1.3.5 Some Good Practices With Symmetrical Key

- A key order must always validate the initial requirement for the key, the number of copies, the time when the key is needed, and the intended recipients.
- Revalidate requirements each time new keys are generated.
- Ensure that the person ordering and the person receiving the key are authorized for the key.
- Do not create and distribute the key too early (i.e., keep the storage time short). There must be enough lead time to ensure that all recipients have gotten their key.
- All key must be securely generated. This includes checks on the created key to ensure randomness.
- Secure local generation may be the best method.
- Key should be securely distributed using benign techniques where available. Where benign techniques cannot be used, limit the number of people having authorized access to the key. Use physical distribution only where absolutely necessary.
- Limit the size of the cryptonet to reduce the number of people who have access to the key.
- Limit the cryptoperiod of the key to limit the damage of an unidentified compromise.
- Limit the amount and duration of contingency key created to reduce the potential for compromise during the storage period.
- Develop procedures to quickly notify all users of a compromised key and how to replace the key with a new one.
- Train users not to use compromised key while waiting for their replacement key.
- Develop effective accounting to track the status of all keys throughout their life cycle.
- Periodically validate all key-handling procedures.
- All procedures and policies must be rigorously enforced.

8.1.4 Infrastructure Directory Services

8.1.4.1 Overview

Infrastructure directory services—through a structured naming service—provide the ability to locate and manage resources within a distributed environment. The directory also provides access control over all the objects represented within this distributed information service. Directory design can be categorized by objects within (scope of content) and functionality (range of services) supported. Within the context of this document, Directory services (see Figure 8.1-10) support provisioning of symmetric and asymmetric key material, as well as the management data for confidentiality, integrity, and identification and authentication across the enterprise.

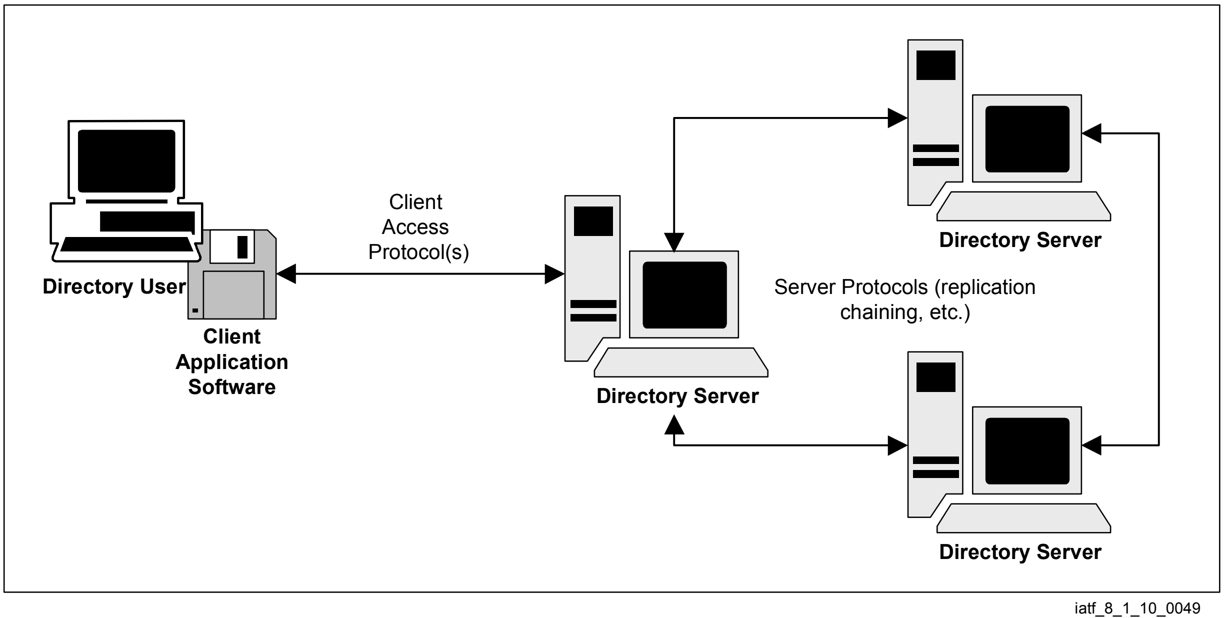

Figure 8.1-10. Directory Model

Infrastructure directory services provide a means to associate multiple elements of information with respect to a specific person or component. This association is managed in a hierarchical organization and indexed by name association. The most common example is a telephone system “white pages” that supports the association of name resolution with address and phone number elements. In the evolving distributed network environment, much more information needs to be managed, requiring more than general-purpose directory functionality. Today, a majority of deployed directory systems are considered “application-specific,” such as PKI, white pages, e-mail, or Network Operating Systems (NOS) directories.

8.1.4.2 Characteristics of Infrastructure Directory Services

Infrastructure directory services have several key characteristics. These characteristics are defined as follows:

- **Defined Name Space.** Directory services typically invoke a hierarchical namespace logically structured in an inverse tree. This naming format can be used to consolidate the accesses, easing user location of information. X.500 distinguished names, Request for Comment (RFC) 822 e-mail naming, and DNS domain names may be used.

- **Highly Distributed.** Directory services reliably distribute the data to multiple servers, whether they are located across an enterprise or within a LAN environment. The mechanisms to allow partitioning of information, its access constraints, and timely access are provided. Additionally, the ability to replicate data across the Directory services makes the system more resistant to failure and maintains accessibility.
- **Optimized Data Retrieval.** Directory services enable the user to search on individual attributes of an object. The design supports a significantly higher ratio of "reads" to "write" operations. Most directory products assume 99 percent of the operations accessing the Directory Information Base (DIB) will be lookups and searches, as opposed to relatively few changes or additions and deletions.

Infrastructure directory services are expected to provide access to any application. Those core applications that will access directories are X.500 Directory Access Protocol (DAP); LDAP; e-mail (S/MIME V3), and a Web-based access (https). Future enhancements will include support for dialup accesses, in support of wireless key management.

The types of clients that access directory services are as follows:

- **Interrogation Clients**—performing general queries for user information.
- **Modification Clients**—performing queries and being cryptographically enabled to perform strongly authenticated binds and modification operations on selected user attributes.
- **Administrative Clients**—who have all the features of the modification client and are permitted to manage user entries and operational information.

8.1.4.3 Information Model

The information model describes the logical structure of the DIB from the perspective of both the directory users and the administrators (see Figure 8.1-11). The information model defines the relationships between the objects, attributes, and associated syntax in a "schema." The user information portion contains the information about a directory object that is viewable by the majority of the accesses to the DIB. The

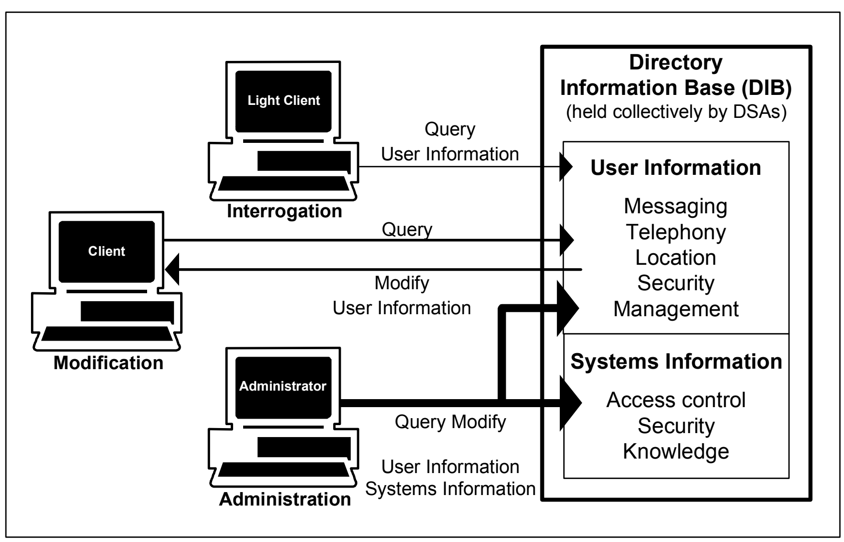

Figure 8.1-11. Directory Use Access

operational and administrative information portion of the DIB contains those elements of information used to track directory operations. These attributes are typically schema information, access control information, and information related to replicating data. Operational and administrative information is not returned in response to normal directory queries.

Further discussions related to directory distribution and Directory System Agents (DSA) information models will be included in future releases of this document.

8.1.4.4 Directory Information Tree

The directory system schema is the set of rules that define how the Directory Information Tree (DIT) is constructed, defines the specific types of information held in the DIB, and defines the syntax used to access the information. A schema has three components:

- **Classes**—the set of objects within.
- **Attributes of each object class**—the set of properties allowed by that class of object.
- **Attribute syntax**—which delineates the syntactic form and any matching rules used with that attribute.

In X.500-based directory systems, an object identifier (referred to as an "oid") references object classes and attributes. In many LDAP systems, the data is essentially a string of characters, with no equivalent object identifier. This is problematic in those environments where compilers are used to interpret the data and apply cryptographic services to that data. The use of Abstract Syntax Notation number One (ASN.1) and associated Distinguished Encoding Rules (DER) is critical to ensuring security mechanisms applied to data in one component or domain will remain intact when used in another component or domain.

These three elements follow a set of rules to ensure appropriate placement of the objects into the DIT. Content rules identify mandatory and optional attributes within a given object class. One problem associated with the use of the X.509 CA object class is that it requires a *userCertificate* attribute. Thus, when the entry for the CA is created, either the CA must have the privilege to create the entry and post a certificate at the same time, or the operation will fail, violating the content rule. Many environments use directory administrators to create entries (add an object class) and allow other entities (like the CA) to populate (add) attributes at some future time. The newer LDAP V2 schema defines a *pkiCA* object class, where the certificate information is optional. Thus, a directory administrator can add the object class, and the CAs can subsequently add the attributes with valid data.

Schema extensibility is a very useful feature to incorporate into a directory system. As new elements of data are defined, they should be added to a directory without requiring the directory to be restarted or the compiler reconfigured. More products are providing this feature; however, if a new object is added to the directory, consideration should be given to the upgrade of the clients that may need to retrieve and use this new element.

A DN is a sequence of naming attributes that uniquely identify an object that may be represented by an entry in the directory. Objects that may be identified using a distinguished name include organizational units, people, roles, address lists, devices, and application entities. A DN is used as the primary "key" to locate an entry in the directory system. The DN is also typically used to identify the subject or issuer of an X.509 public key certificate.

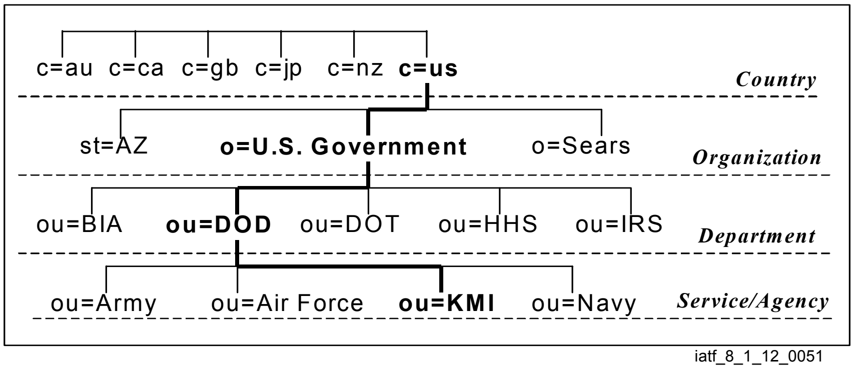

Figure 8.1-12. Key Management Infrastructure Directory Information Tree

The naming attributes that form a DN are organized in a hierarchy reflecting the DIT with a name lower in the tree identified relative to its parent entry by adding Relative Distinguished Name (RDN) attributes to the parent's DN (see Figure 8.1-12). Note that naming conventions and registration processes must be clearly articulated for a domain. Before an entry is created for an object in the directory (or a certificate created for that object), it must be allocated a DN that is unique across the enterprise. An RA normally performs the creation of a distinguished name in the directory system. Disambiguation of names is critical for key management functions; however, it is usually approached with an emotional perspective rather than a logical view. Recommendations for namespace management will appear in later versions of this Framework.

8.1.4.5 Security Model

The security model defines the access control framework and identifies mechanisms for the access control scheme applied to a DIT segment. A comprehensive security model not only addresses user access to the information within the DIB, but also includes access controls on the application itself. In addition, the security model should include the management of the cryptographic keys for identification and authentication (I&A) and, if appropriate, confidentiality for the directory servers. The confidentiality services in an infrastructure directory system are typically applied at the network or transport layer.

The security services defined below are considered against the three general threats of unauthorized disclosure, unauthorized modification, and unavailability of information contained in a directory system. The information is vulnerable when held within a DSA or when transiting elements of the directory. The security services are as follows:

- Authentication.
- Access Control.
- Confidentiality.
- Integrity.

Authentication

Peer entity authentication is performed between the clients and DSAs and among DSAs to provide corroboration that a user or entity in a certain instance of communication is the one claimed. The authentication mechanism can be a name and password, or an exchange of cryptographically bound credentials, referred to as strong authentication. Strong authentication relies on the use of asymmetric encryption. Asymmetric encryption uses the combination of a public component and a private component to sign digitally the credentials of the user or entity authenticating itself to the system. A digital signature guarantees the origin and integrity of the information that is digitally signed. This binding of the public key and its holder's identification information is conveyed through an X.509 public key certificate that is generated by a CA. The generation of these identity certificates is usually within the bounds of an organization's certificate policy. Within a CPS, procedures should be used to create, maintain, and revoke credentials for the clients, managers, and directory servers themselves.

It is sound practice for all DSAs to be able to process bind requests that are name and password based, as well as strongly authenticated, using an agreed on digital signature algorithm. DSAs should support an access control policy that prevents the unauthorized disclosure or modification of information based on the authentication level used. The DSA should strongly authenticate itself to its communication peer (i.e., DSAs, clients, and management entities) as required by policy. The success or failure of the steps in the authentication process should be audited and stored in the DSA audit database to facilitate compromise recovery and to enhance security of the directory.

Additionally, the DSAs should not permit access to any information until all access control checks have been performed and granted. DSAs should support a standards-based (Internet Engineering Task Force [IETF], RFC 2459) signature validation process. This process should include validating the CA that produced the certificate used to sign the I&A information (i.e., validate the certification path). If the path validation process cannot be completed, DSAs should reject the request and generate an audit notice. Additionally, the DSA may lock out the user from any subsequent accesses.

Once the communications partners have successfully authenticated themselves to each other, the DSA should be capable of limiting access to information stored within its DSA according to the parent (host) system security policy. The DSA should constrain setting access and privileges to authorized management entities only.

Access Control

Access control is based on a relatively simple concept: either a list of users and the permissions to which they are entitled, or a list of protected items and the permissions necessary to access them, is held within the directory. This information is contained within access control information (ACI) items. ACI items can be held within a number of parts of the directory depending on their intended usage and sphere of influence.

The Access Control Decision Function (ACDF) specifies how ACI items should be processed to determine whether access should be granted for a particular operation. Figure 8.1-13 is based on the ISO/IEC 10181-3 Security Framework in Open Systems standard (Part 3—access controls). The ACDF decides whether to grant or deny access to the requested object(s) by applying predefined access control policy rules to an access request.

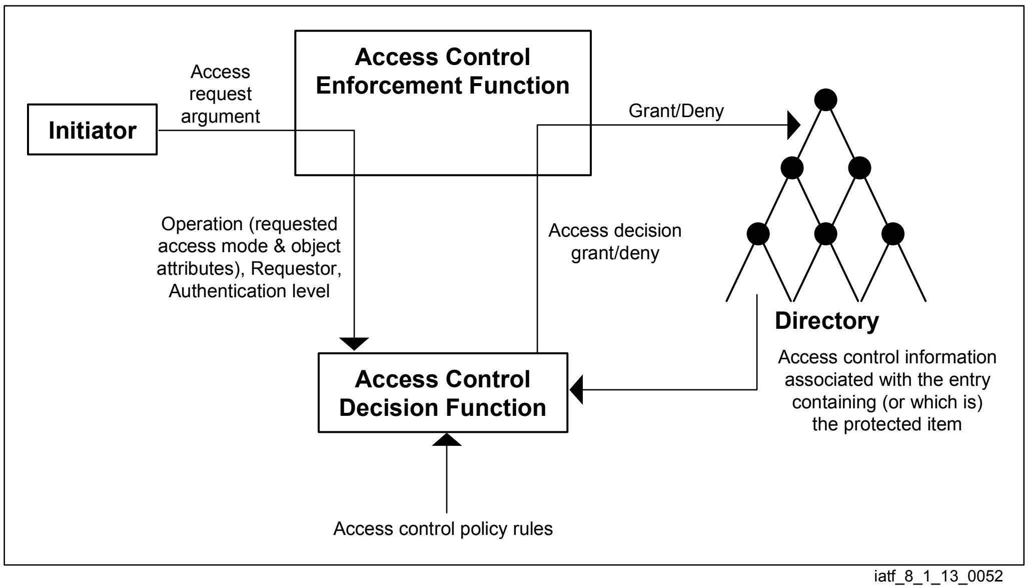

Figure 8.1-13. Access Control Decision Function Required for Access Control

In some situations, the directory may not give sufficient assurance that data is kept confidential in storage, regardless of access controls. Confidentiality of attributes in storage may be provided through use of an encrypted attribute. Variations are defined in ITU-T X.501 (1997) and in emerging IETF standards. In all instances, the directory servers do not support the encryption and decryption of this information.

Confidentiality

Confidentiality at the application layer is an extremely difficult service to provide. It is defined in the 1997 X.500 Series of Recommendations, but relies heavily on the General Upper Layer Security (GULS) and the use of the Open Systems Interconnection (OSI) Presentation Layer. At this point, there are no directory server products that support this service. Emerging standards permit the use of the Transport Layer Security (TLS) with LDAP, yet again, there are few, if any, products that support this service. Network and transport layer security is an extremely useful part of the layered security approach for a directory service.

Integrity

If integrity is required on information stored in the directory, the information should be signed. The user who requires validation of the integrity of that information should validate the signature to ensure no unauthorized modifications have occurred. If an attribute requires integrity, the syntactical definition should expressly define it as a signed object. In the public key schema context, certificates and CRLs are signed objects.

The ability to support signed operations on all operation requests received and to generate signed responses to those arguments, needs to be evaluated against a performance, risk analysis and policy basis. In many cases, it is less complex and equally secure to invoke a secure channel at the network or transport layer in conjunction with the initial binding operation. Part of the security management requires the integrity protection to be negotiated and agreed-on when establishing connectivity.

Any of the information stored within a Security Management Information Base (SMIB) should be protected against manipulation or destruction by unauthorized users or end entities. Changing any of the thresholds associated with collection of audit information should be made available to only authorized audit management entities. When information from one domain is replicated into another domain, the agreement to shadow should contain details on how archive of and access to audit data will be supported. Further details with respect to this critical security feature will be provided in later versions of this document.

8.1.4.6 Credential Management

Directory servers will require their own identity credentials when they digitally sign bind operations or other operations that may require integrity. Strong authentication is not widely deployed, but when it is, the volume of signature verifications requires either a "bank" of card readers, with duplicate hardware tokens in each reader, or some form of hardware accelerator deployed on the server hosting the directory service.

Directory Administrators (DA) will use their own sets of credentials when logging into the directory server. This permits auditing and tracking of those actions taken by the DA when modifying any of the operational information. DSAs will use their own credentials when responding to strongly authenticated bind requests, and when initiating strong binds between DSAs. In the few cases in which cryptographic services are enabled in directory systems, the credentials are usually uploaded to the DSAs through a floppy interface or via a PCMCIA bus interface. The initial keying and subsequent rekeying of hardware accelerators will be discussed in future versions of this document.

8.1.4.7 Implementation Considerations

The directory service must have realistic performance characteristics. Performance can be measured in a number of ways: ease of use, robustness, timeliness of service restoration, and speed of access response. These aspects of the system and the generation of domain specific concepts of operations (CONOPS), policies, and procurement procedures will be discussed in later versions of this document.

- Ease of use is a factor of the system design and the tools presented to the directory user such as click and point, icons, windows, scripts, and status messages.
- Robustness deals with product and system reliability and integrity. Again, these will have to be specified in terms of integrated logistics support (ILS) and life-cycle costing (LCC) needs and in terms of mean time between failure (MTBF) or mean time to repair (MTTR) type specifications.
- The availability goal is to provide availability of any directory service 24 hours a day, 7 days a week. In the certificate management context, revocation information must be available on demand.
- Service restoration deals with the recovery time for a single DSA to attain an operational state after switching on or switching the clients (and other attached DSAs) to an alternate DSA. This should not exceed 5 minutes if the DSA is in a strategic environment. In a tactical environment, it should be less than 1 minute.
- For defining the speed of response requirements, the directory system can be seen to provide two types of access characteristics: the human access requirements, which deal with information retrieval (such as white pages information) via a man-machine interface, or specific system functions, which need to resolve, for example, names to addresses for message routing. This interface is considered to be a machine-to-machine interface. Both of the above have performance requirements. However, how these are characterized and presented can be quite different. Underlying the performance of such a large-scale system is naturally the individual DSA performance and the links used between them to other DSAs and the accessing clients.

8.1.4.8 Client Caching Guidelines

Employing client caching is a matter of domain policy. However, the guidelines below may be followed, especially for clients caching certificate-related information.

- Store cached information in nonvolatile memory.
- Treat cached entries and cached certificates separately for the purpose of determining the useful life of the cached information. Extend the useful cache period for the certificate, because it is a relatively static entity with its own expiration time and revocation procedures.
- Capture and record, with the cached entry, the date and time that an entry was last obtained in order to determine the expiration time of that entry.

- Upon receipt of a CRL, all components containing cached certificates compare the cached certificates against the list of revoked certificates and purge those cached certificates matching the certificates listed in the CRL.
- Purge a cached certificate upon the expiration date.

8.1.5 Infrastructure Management

The KMI/PKI infrastructure has many of the same characteristics and issues as the certificate management and symmetric key generation subscriber services described in Sections 8.1.2, Certificate Management, and 8.1.3, Symmetric Key Management. However, it is also a much more attractive target because a successful attack potentially subverts the security of a large number of subscribers instead of only one. In addition, it has a number of additional requirements and responsibilities not associated with subscriber services, which introduce potential new vulnerabilities. Because of these increased security concerns, the design of a KMI/PKI needs to address a wider range of issues than just supplying keys or certificates to subscribers. Although the technological solutions for these problems are substantially the same as those described in Sections 8.1.2, Certificate Management, and 8.1.3, Symmetric Key Management, their implementation, layering, and procedural security solutions will be more robust. The basis of managing a secure infrastructure is trusted personnel performing their duties correctly. This section focuses on the procedural issues involved in managing the infrastructure. It discusses unique technical requirements and issues involved with designing, developing, and operating a secure infrastructure as appropriate.

This section assumes a PKI-based infrastructure with a “trusted” root element (ROOT CA) acting as a domain’s signing authority. The root element will be the basis of the domain’s trust relationship among subscribers. The root will enroll authorized infrastructure elements (e.g., subordinate CAs). These authorized elements must ensure that they enroll only other infrastructure elements that they trust. Finally, the CAs will properly identify each subscriber they enroll and ensure that their certificates are correct. The domain’s trust relationship allows subscribers to believe that the information contained in validated certificates is correct.

Building and operating an infrastructure’s trust relationship involves much more than just issuing certificates to the CAs. The KMI/PKI also has to manage itself. This requires the KMI/PKI to develop and enforce acceptable security policies and procedures, manage the key and certificate process to ensure that each element is operating correctly, manage the domain’s external relationships (e.g., determine acceptable cross-certification requirements), and ensure availability. Unique KMI/PKI management requirements include the following:

- Policy creation.
- Policy enforcement.
- Key and certificate accounting.
- Compliance audit.
- Cross-certification.
- Operational requirements (e.g., training, physical, personnel, operating procedures).
- Disaster recovery mechanisms.

All PKI security attacks defined in Section 8.1.2, Certificate Management, apply in equal measure to the infrastructure itself. However, the consequences of the attacks are now greater, and the infrastructure also has to protect itself against a number of new attacks that target its management of the subscribers' keys and certificates. Examples of attacks are as follows:

- Deny global service by taking down portions of the KMI/PKI.
- Substitute attacker's public and private material for KMI/PKI element's material to control the issuing process of subscriber's certificates.
- Destroy the domain's trust relationship via the incorporation of inappropriate elements within the KMI/PKI (e.g., inappropriate cross-certification link).
- Compromise the data recover infrastructure.

Although an attacker could theoretically attack the infrastructure to obtain access to an individual subscriber's information, a more likely scenario is an attacker trying to subvert the infrastructure to gain access to information on a large number of subscribers. This makes the security requirements on the internal KMI/PKI certificates stronger than on the equivalent subscriber's certificates. These increased requirements might the following:

- Higher assurance in the identification process for KMI/PKI elements.
- Higher assurance in generating keys and certificates for KMI/PKI elements.
- Better protection against compromise.
- Increased mechanisms for the detection of potential compromises.
- Rigorous personnel/physical/procedural security measures.
- Stronger security architecture for limiting and monitoring operator actions.
- Stronger data recover security.

8.1.5.1 Policy Creation and Management

One of the most important aspects of establishing and maintaining a trust relation for a KMI/PKI is its security policies. To establish the trust relationship within the domain (and other cross-certified domains), the policy must provide a basis for the subscribers to know and understand the degree of security that the KMI/PKI actually gives them. No KMI/PKI can guarantee that it is totally secure and that there is no possibility that there are unauthorized subscribers. Subscribers must know to what degree they want to accept the KMI/PKI's assurance that the other subscriber with whom they are communicating is the person identified in the certificate. The only way that a subscriber can determine what trust to place in the domain is by examining the KMI/PKI's security-related policy. KMI/PKIs must document their policies for both subscribers' keys and certificates and their own internal keys and certificates. Depending on the trust requirements for the specific application, these policies may range from very tight to fairly loose. Section 8.1.6, KMI/PKI Assurance, discusses how to define policies for applications with different levels of security requirements.

The approach to defining security policies for KMIs and PKIs tends to differ in that PKIs are implemented in a global federal, intragovernment, and commercial community, whereas KMIs tend to be operated in smaller national security communities. Consequently, considerable effort has been devoted to developing international standards for PKI certificate policies, whereas KMI security policies tend to follow more local and national intergovernmental standards.

The ITU X.509 standard describes a Certificate Policy as follows:

> "…a named set of rules that indicates the applicability of a certificate to a particular community and/or class of application with common security requirements." [1]

An IETF informational RFC (PKIX 2527 Certification Policy/Practice Statements [5]) that defines a framework for developing policies can be found at http://www.ietf.org/html.charters/pkix-charter.html. The policies cover a wide range of issues, from defining the rules for initializing a new infrastructure element or subscriber, to the physical and personnel requirements for the domain, to what happens in an emergency. The Certificate Policy addresses issues such as the following:

- Certification identification requirements.
- Key generation (subscriber/infrastructure, hardware/software, etc.).
- Procedural security requirements.
- Computer security requirements.
- Physical and personnel security requirements.
- Operational policy requirements.
- Requirements on subscribers (e.g., protect key).
- Interoperability requirements (e.g., cross-certification).
- Rekey mechanisms.
- Key and certificate distribution.
- Certificate profile.
- Network security requirements.
- Compromise recovery requirements.
- Liability discussion.
- Types of applications in which the certificate may be used.

Developing Certificate Policies to the IETF Framework has proven extremely valuable in allowing an "apples to apples" comparison of PKI security practices. The IETF 2527 document has become the basis for numerous other Certificate Policy management and evaluation standards worldwide.

Certificate Policies affect the relying parties, subscribers, and those developing and deploying PKIs. They are also the basis for achieving "policy interoperability" among interoperating PKIs. Therefore, the Certificate Policy Management Authority (PMA), or Policy Authority, should consider the interests of all these parties when composing and reviewing the Certificate Policy. Furthermore, because public key certificates are often planned for use in applications having legal requirements (e.g., financial transactions), legal counsel must be an important part of most Certificate Policy development efforts.

Once created, there are numerous further actions that are necessary to make a Certificate Policy useful. The infrastructure's approach to meeting the Certificate Policy requirements must be documented in one or more CPSs.

Certificate Policies should state high-level security requirements and leave implementation descriptions to lower level documents, such as CPSs. In many ways, the relationship between a Certificate Policy and a CPS is analogous to that between a Request for Proposal (RPF) and a proposal. Authors of a Certificate Policy and an RFP strive to limit their statements to functional or security requirements and not to define specific implementations. Authors of proposals and CPS documents strive to describe specific implementations and need to avoid simply repeating requirements. The PMA is responsible for reviewing CPS documents to ensure they meet the PKI's Certificate Policy requirements.

The CPS documents should be distributed to the PKI elements responsible for fielding and operating the PKI. The KMI/PKI components are procured or designed to the specifications of the approved CPS implementation document, and personnel are trained in the procedures defined in the CPS. During operation, the KMI/PKI must employ mechanisms to enforce—and document—that the CPS provisions are followed correctly by the PKI. Usually, such enforcement consists of a regime of compliance audits conducted by third-party auditors (or other professionals).

Finally, the policies should be periodically reviewed, updated, and distributed to ensure that they still provide the necessary security. Without these actions, the subscribers have no idea how much trust to place in a key or certificate.

Attacks against the policy creation process can disrupt the domain's trust model by misrepresenting the level of security provided by the KMI/PKI. Although this misrepresentation does not lead to any direct attacks against either the KMI/PKI or the subscriber data, it may permit the key or certificate to be used in inappropriate applications where other attacks may be successful.

8.1.5.2 Registration

Subscribers typically "trust" the local element that provides their key or certificate because in a normal office environment, the local operator is often someone known to the individual. The subscriber also generically "trusts" the KMI/PKI root, which might be the company personnel office. The KMI/PKI trust relationship relies on the fact that every other infrastructure element—and by inference every other subscriber—is just as reliable as those elements which the subscriber personally trusts. Cross-certification extends the trust relationship to all infrastructure elements in all the other cross-certified domains.

The abilities to approve new CAs and to cross-certify other domains are critical functions that must be strictly limited. Registration is the procedural process for identifying to the infrastructure the people and elements authorized to change the domain's trust relationship. For infrastructure elements, there are normally two separate processes involved. The first reviews the policy implications of adding a new infrastructure element or allowing cross-certification. This is a procedural process done out-of-band by the Certificate PMA. The second process is to implement the policy decision by creating the appropriate certificates. The persons responsible for implementing the decision are the root and CA operators.

When a domain establishes an infrastructure, it will identify the root and CA operators. The CPS (Section 8.1.5.1, Policy Creation and Management), should outline the qualifications, such as clearances and training, for the personnel who assume these roles. The operators must also be registered with the software being used within the system. These operators normally have special accounts for access to the administrative functions at each component. To access these accounts, the operators will need to authenticate themselves to the components through the use of passwords, public key certificates, or hardware tokens. The components need to ensure that these authentication processes are strong enough that an attacker cannot gain access to these special functions. The effect would be that the attacker could enroll an infrastructure and hence unauthorized subscribers.

8.1.5.3 Ordering and Validation

The ordering process within the infrastructure consists of two phases: making a request to the registered authority to add a new infrastructure element or cross-certification, and providing the necessary information to generate the certificate (e.g., CA's identity, CA's public key) in a secure, authenticated manner. The ordering process validates the request and provides a mechanism for protecting the integrity of the public key and authentication information. The generation process will bind the authentication information into the certificate.

Although the electronic ordering mechanisms discussed in Section 8.1.2.3, Infrastructure Processes, can establish new KMI/PKI elements, because of the sensitivity, an off-line manual process is more likely. Complicating the issue is the possibility that in many domains, the new element will not be in physical proximity with its superior element. In this situation, the enrolling CA will not be able to personally identify the ordering CA.

Although subscriber's orders require only validation of their identity and the correctness of their certificate information, an infrastructure element must show that they properly implement the domain's policy. This requires that before the KMI/PKI generates a certificate for an infrastructure element, (1) it establishes the need for the new element with its specific set of privileges, (2) the element understands the policy and complies with its requirements, and (3) the people who are operating the element are trustworthy.

Cross-certification is also likely to be an offline manual process. However, it is likely that the two domains will not be in close physical proximity and will not be able to rely on personal identification. Before generating a cross-certification certificate, the KMI/PKI must validate the request. Beyond establishing the identity of the domain and its certificate information, this requires that the KMI/PKI establish the need for a cross-certification with this particular domain, determine that the policies of the two domains are consistent, and ensure that the other domain complies with its stated policy.

8.1.5.4 Key Generation

Please refer to Sections 8.1.2.3, Infrastructure Processes, and 8.1.3, Symmetric Key Management. No unique infrastructure requirements exist. Given the additional threat against the infrastructure, it needs a higher degree of assurance in the keys, which can take the form of longer keys, hardware key generation and storage, or input from multiple elements.

8.1.5.5 Certificate Generation

Once a new CA is authorized, the technical process of creating and signing a certificate for the infrastructure and the subscribers is similar to the process for subscriber certificates (Section 8.1.2.3, Infrastructure Processes). The primary difference is that the infrastructure must generate the initial root key and certificate in a unique way. Certificates for the other KMI/PKI elements and subscriber are identical. Some differences may also exist in the certificate's profile, however, because some of the X.509 v3 certificate fields apply only to the infrastructure and some apply only to the subscribers.

The root certificate is unique because it is self-signed; therefore, no higher level device exists that can generate the certificate. This creates a unique process in a security-critical function. The root performs the following activities to initialize the domain.

- Create the domain's cryptographic parameters (when required).
- Output the domain's cryptographic parameters in order to distribute them to the subscribers.
- Generate a public and private signature key.
- Generate a root certificate signed with the private signature key.

The biggest difference in the certificates is that infrastructure certificates populate the constraint and policy fields to limit the ability of a compromised KMI/PKI element to affect other elements. The generation process must ensure that the certificates are appropriate for the certificate's application. The specific fields populated depend on the domain's policy. The federal PKI certificate profile, which can be found at http://csrc.nist.gov/pki, identifies the following profile: [6]

The certificate profile identifies four types of certificates with different requirements: root, general CA, cross-certificate, and end subscriber. All types of certificates use the complete set of X.509 base certificate fields except issuerUniqueIdentifier and subjectUniqueIdentifier. The various certificates differ in the extension fields. The root certificate populates only two extensions: subjectKeyIdentifier, which identifies the specific root key being used, and basicConstraints, which identify it as a CA. The CA and cross-certification certificates are similar. They must process (although not necessarily use) all extensions except privateKeyUsagePeriod and subjectDirectoryAttributes. Three fields—policyMapping, nameConstraints, and policyConstraints—used in infrastructure certificates are not used in subscriber certificates. The profile identifies other differences in the specific fields for each extension.

The root private key is the most valuable key in the domain. If compromised, the attacker can create unauthorized certificates that allow him to masquerade as anyone in the domain. Because the root certificate is self-signed, it is uniquely vulnerable to substitution attacks. If an attacker can get a subscriber to believe that the subscriber's self-signed certificate is from the root, then the attacker can issue certificates that the subscriber will believe are valid. Also, if an attacker can force the root to use a known key or generate a key susceptible to cryptographic attack, then they can generate their own root certificate. Also, it is likely that there will be a stored copy of the signature key in case of a root failure. If the root fails and there is no signature backup, the entire domain must be reinitialized with the new root certificate. These security issues highlight the extreme care that the infrastructure must take to protect the root key and any copies that might exist.

8.1.5.6 Distribution

The KMI/PKI must ensure that all subscribers in the domain have authenticated access to the necessary system information and certificates. The directory discussed in Section 8.1.4, Infrastructure Directory Services, will be one method of distribution of certificates and other parameters. The infrastructure has to distribute four items: the system parameters, its own certificates, compromise recovery data, and subscriber certificates.

The authenticated delivery of the system parameters, including the domain's cryptographic parameters (when available) and the root certificate, are security critical because they are the foundation of the domain's trust relationship. Although they are public values, their authenticity is critical to the correctness of the subscriber's certificate validation process. The parameters, created by the root during system initialization, are used by the CAs during the generation of other certificates. Distribution mechanisms may include a directory, off-line distribution, or local distribution through the CA. The KMI/PKI must also ensure that all subscribers have authenticated access to its certificates and compromise recovery information.

After certificate generation, the KMI/PKI provides the subscriber with certificates. Before activating a new certificate, the infrastructure and subscriber should check that the certificate was generated properly. The infrastructure must check that the certificate owner has access to the private key that corresponds to the certificate's public key. Proof of Possession (POP) is one protocol solution for performing this check. The subscriber must check that the certificate contains the correct public key and subscriber information. After completing the checks, the subscriber indicates that the infrastructure should post the certificate.

8.1.5.7 Accounting

The KMI/PKI has to be able to track the location and status of keys and certificates throughout their life cycle. There will likely be a requirement to archive the accounting information because of the legal need to be able to document the life history of a key or certificate for as long as the signature might need to be verified. The accounting information for each certificate should provide, at a minimum, the certificate contents plus the applicable information for each task, including the following:

- Task.
- Time.
- Status (completed/error).
- Operator involved.
- Element that originated the task (e.g., where did the order originate).
- Other element(s) involved in the task.
- Acknowledgment from other element(s) involved.

Accounting has real-time security and administrative requirements. It provides a security service by allowing the check that each step of the process was proper (e.g., the certificate generation process checks the status of the order validation) before the beginning of the next task. Accounting also tracks the interaction between various components by requiring each element to acknowledge to other involved elements that it has completed its portion of the processing.

The primary use of an account is administrative. The system needs to be able to track the history of keys and certificates in case of future challenges to its authenticity. Accounting is useful for the following tasks:

- Showing an outside observer the infrastructure life cycle for any key.
- Proving to an outside auditor that the policies and procedures were followed correctly.
- Providing damage assessment of operator actions if an operator is subsequently shown to be untrustworthy.
- Recording certificate information from the ordering process.
- Archiving a key’s history.
- Archiving a token’s history.

Depending on the KMI/PKI architecture, a single element or many elements can perform the accounting. All accounting records must be protected against accidental deletion or modification, or malicious attacks. If several elements perform accounting, either for one key or certificate or because multiple certificates from different elements reside on one token, there is an additional issue of coordinating the partial accounting records into a complete, authenticated set of records.

8.1.5.8 Compromise Recovery

An infrastructure element can compromise either its signature key or key agreement key. The compromise of a KMI/PKI element's key agreement key is the same as for a subscriber's key (Section 8.1.2.3, Infrastructure Processes).

Because the compromise of an infrastructure element's signature key invalidates all lower level certificates that include the element in their validation path, it is the more serious problem. This includes not only the direct certificates it created for lower level CAs and subscribers but also any certificates created by the CAs. It is critical that the infrastructure be able to reenroll the affected elements and subscribers quickly and painlessly, while removing any unauthorized subscribers enrolled by the compromised element. The infrastructure must be able to inform the subscribers and cross-certified domains about an infrastructure compromise quickly and accurately, while rapidly rekeying the affected elements and subscribers. The responsibility for informing the subscribers resides in the element that enrolled the compromised element. The mechanisms for notifying subscribers about the compromise of an infrastructure certificate are the same as those defined in Section 8.1.2.3, Infrastructure Processes, a CRL or online verification.

For a compromised root, the same mechanisms theoretically work, but it is unclear whether the applications support will be there. Possible solutions include placing the root certificate on a root generated CRL, placing the root certificate on the PCA CRL, or performing online verification. When checking a CRL, normal processing is to look for the certification on the CRL from the enrolling CA. Both possible CRLs for the root (its own or from a subordinate CA) are exceptions to this processing, and it is unclear if the applications will support them. Online verification protocols are still in the design stage, and it is unclear if they will report the root as compromised. Alternative workarounds, such as placing every CA on the appropriate CRL, may meet the requirement.

The recovery process for reenrolling subscribers is straightforward, but the process must be performed quickly to minimize the impact on the subscribers. Starting at the compromised element, it generates a new public and private key pair and a higher level element generates and signs and distributes the new certificate. Once the element is operational again, it can begin to reenroll its subscribers. The reenrollment process requires a revalidation of every subscriber, using any of the mechanisms outlined in Section 8.1.2.3, Infrastructure Processes. An issue is how to deal with the occasional PKI subscriber who has not tried to validate a certificate since the compromise. Subscribers will not realize they need to be reenrolled. The infrastructure can allow them to continue to have an unusable certificate, or it can contact them about being reenrolled. Lists of subscribers should be available from either the local accounting records or the directory.

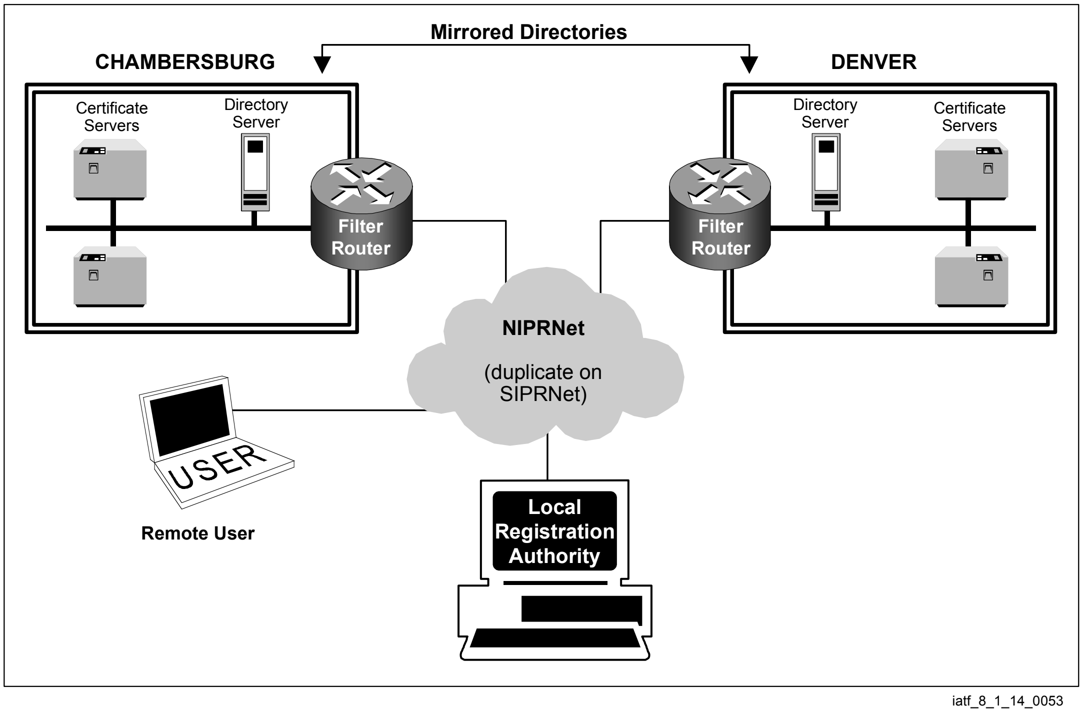

Figure 8.1-14. DoD Class 3 PKI Architecture

8.1.5.9 Rekey

An infrastructure element's rekey process differs for key exchange key and signature key. An infrastructure element's key exchange key is similar to a subscriber's rekey addressed in Section 8.1.2.3, Infrastructure Processes. Signature rekey has major effects on the CAs or subscribers created by the element; therefore, the KMI/PKI must give strong consideration to how often it will rekey the infrastructure elements. The consequence of rekeying an infrastructure element's signature key is that every certificate in its verification chain must also be rekeyed. This action creates a tradeoff between security and subscriber friendliness over the frequency of rekey. Security considerations push for frequent rekeys because of the consequence of an undetected compromise or a crypt-analytic attack of an infrastructure element. Subscriber friendliness demands infrequent rekeys because of the impact on the subscribers of rekeying the infrastructure.

The security tradeoffs are straightforward. The private signature key of an infrastructure element is a high value target because a compromise allows an adversary to masquerade as anyone in that element's domain. The longer the key remains in use, the greater the incentive for attacking it, and the better chance the adversary has of being successful. Once the element is rekeyed, the old signature key has no value.

Infrastructure rekey operational issues that should be included in the process are listed below.

- There should be a graceful rollover to the use of new keys without a period of community isolation or noninteroperability.
- Revocation notification must be maintained during the rollover. This means that KMI/PKIs will probably maintain multiple, simultaneously current CRLs.
- Note that a CMA may continue to sign CRLs with the old key, long after it has ceased signing certificates with that key and until the last certificate signed with that key expires and its CRL-inclusion period passes.
- It should be possible to issue certificates that will not fail validation because of expired signing authority certificate (i.e., the requested certificate should verify for a reasonable time period even when issued just before rekey of the signing authority). (This action is often accomplished by making the signing authority certificate validity period longer than the signing authority private key usage period.)
- The issuance of certificates should not be unreasonably delayed when authority rekey is pending—that is, an end subscriber certificate request should not kick off an authority rekey, possibly extending to multiple levels of the hierarchy, for which the subscriber must wait.
- The mechanism will have to live within the constraints of the cryptographic token(s) employed at the time of its introduction.

One method of minimizing the subscriber impact is to use the current key to authenticate the new key. The steps to initiate this action are listed below.

- The Root CA generates a new key and creates a new certificate with its public key signed with the current signature key.
- The Root CA also creates a new certificate for the current key and signs it with the new signature key.
- Subscribers needing the old CA certificate containing the old key must cache it locally because it will not be available from the directory.
- All subordinate subscribers and authorities should be notified of the impending rekey so that they can cache the certificate containing the old key and, probably, the last CRL signed by the old key.
- Applications must recognize when data is signed using a private key associated with an old certificate and obtain the old certificate from its cache.
- Applications may have to forego checking of current CRLs issued by the rekeyed authority and incur the associated risk.
- All subscribers and authorities whose certificates were signed by a rekeyed authority should obtain as quickly as possible new certificates, signed by the new key.

CAs will continue to issue CRLs signed by the old key until one CRL-inclusion period after the expiration of all certificates they have issued. Therefore, subscribers can continue to be notified of revocations of certificates signed by the old key.

When it is time for the Root CA to rekey, the subscriber can validate the signature regardless of which key the sender and recipient have. For example, if the Root CA and the sender have both been re-keyed but the recipient hasn't, the recipients validation chain would be as follows: the sender, its CA(s), the new Root CA certificate, and finally the new Root CA certificate signed with the old signature key. Once the Root CA begins its rekey process, each CA can use a similar process to generate its new keys.

8.1.5.10 Destruction

Please refer to Section 8.1.2.3, Infrastructure Processes.

8.1.5.11 Key Recovery

There are two separate issues about key recovery in the infrastructure. The first deals with how KMI/PKI elements perform key recovery. The second deals with the issues involved in developing a key recovery infrastructure.

Key Recovery for KMI/PKI Elements

There are no easy answers about the requirement for key recovery in infrastructure elements. The requirement depends on the policy of the domain. This section defines some of the tradeoffs in the key recovery policy.

In general, signature keys do not need key recovery. The signature key serves no law enforcement purpose and the subscriber suffers no great inconvenience in getting a replacement signature key. Within the infrastructure, however, the enormous impact of rekeying the element and its subscribers for lost or destroyed keys (Section 8.1.5.9, Rekey) drives the requirement for key recovery of certain signature keys. The policy can limit key recovery to only certain elements. Even if some elements, such as the root, require key recovery, other elements within the infrastructure do not. Given the obvious security ramifications of storing signature keys, a robust recovery system must be in place to protect keys against all unauthorized access. The key recovery policy for KMI/PKI element's key agreement keys is the same as for any other domain subscribers.

Key Recovery Infrastructure

There is no one key recovery infrastructure. Either the certificate management infrastructure or a completely separate infrastructure can perform key recovery. The regular certificate management infrastructure would store encrypted keys at the CAs. Advantages include no additional people with access to the key and lower cost, and infrastructure employees could already exploit the keys through other attacks. A separate infrastructure could use any approved method. Advantages include potentially tighter security for the keys and no political fallout for the certification management infrastructure. The next section describes a generalized recovery architecture based on the draft Key Recovery Federal Information Processing Standard (FIPS).

Generalized Key Recovery Model

The key recovery system model defines the minimal set of system components needed to perform key recovery. The key recovery system model is a generalized model that supports a wide variety of different key recovery techniques and data applications. The key recovery system model contains the following components, as a minimum:

- System A (Encryption-enabled).
- System B (Encryption-enabled).
- Recovery Information Medium (RIM).
- Key Recovery Requester System (Requester System).
- Key Recovery Agent(s) (KRA).

The model depicts a key recovery system capable of creating key recovery information (KRI) and recovering the key from the KRI.

The three components—System A, System B, and the KRI medium—collectively define the "Key Recovery Enablement Process." The process also includes an encrypted data medium and a key distribution medium. The encrypted data medium and key distribution medium are the "locations" where the encrypted data and data encryption key are stored or communicated, respectively.

The process of encrypting data and creating KRI is divided between one or more encryption-enabled systems, denoted in the key recovery system model as System A and System B. An encryption-enabled system can encrypt and decrypt data. System A, System B, or both need the ability to create KRI. However, the key recovery system model does not prescribe which system or systems must have a key recovery capability. The RIM maintains the KRI produced by these systems. The RIM may exist over multiple "locations", and may be in the same or different location from the encrypted data and key distribution mediums.

The RIM represents the "locations" where the KRI is stored or communicated, such as a storage device or a communications channel. The key recovery system model does not prescribe how or where the KRI must be stored or communicated, so long as the RIM is available. To allow interoperability between various key recovery schemes, a standard format for KRI on the RIM is essential. Each scheme has a distinct set of information that must be present in order to allow key recovery. A key recovery field (KRF) contains this information. To ensure the integrity of the KRF, the association of the KRF with the encrypted data, and to provide the identities of the key recovery scheme in use and the appropriate KRAs, a key recovery block (KRB) contains the KRF.

The KRI itself is managed or handled in a variety of ways. It may exist for only a brief time during electronic transmission, or it may exist for a relatively long time on a storage device.

The Requester System and the KRA form another subportion of the key recovery system model called the Key Recovery Process. The Requester System and KRAs handle the process of recovering a key from the KRI. They access the encrypted data medium and the RIM and interact with one or more KRAs using a Requester System to recover a cryptographic key from the KRI.

A recovered key can then be used to recover the data, either directly or indirectly, using a Data Recovery System. The data encrypting key is recovered directly when the recovered key is the same key used to encrypt the data. Indirect key recovery occurs when the recovered key is a key encrypting key used to decrypt or recover the data encrypting key.

Requirements

This section defines some of the security requirements on a key recovery infrastructure and its elements. It discusses a high assurance commercial-level recovery infrastructure. Depending on the application, higher or lower assurance infrastructure may be appropriate.

Key Recovery Agent Requirements

- **Cryptographic Functions**—All cryptographic modules shall be FIPS 140-1, Level 3 compliant.
- **Cryptographic Algorithms**—The key recovery scheme shall be at least based on using only FIPS algorithms. The implementation of these algorithms shall conform to the applicable FIPS standard(s) (Same as Level 1).
- **Confidentiality**
  - The KRA shall protect KRI stored against disclosure to unauthorized individuals.
  - The KRA shall protect KRI transmitted against disclosure to parties other than the requester(s).
  - The KRA shall prevent any single subscriber or mechanism from compromising the confidentiality of the KRI.
- **Audit**
  - The product/system shall be able to generate an audit record of the following auditable events.
    - Start-up and shutdown of the audit functions.
    - All auditable events as defined in the system security policy.
  - Examples of auditable events include the following.
    - All requests to access subscriber authentication data.
    - Any use of the authentication mechanism. The authentication information shall not be stored in the audit trail.
    - All attempts to use the subscriber identification mechanism, including the subscriber identity provided.
    - The addition or deletion of a subscriber to or from a security administrative role.
    - Requests, responses, and other transactions generated by the product/system.

- ➢ Requests, responses, and other transactions received by the product/system.
- ➢ The invocation of the nonrepudiation service.

– The audit event shall include identification of the information, the destination, and a copy of the evidence provided. The event shall exclude all private and secret keys in encrypted or unencrypted form.
– The product/system shall be able to associate any auditable event with the identity of the subscriber that caused the event.
– The product/system shall be able to generate a human understandable presentation of any audit data stored in the permanent audit trail.
– The product/system shall restrict access to the audit trail to the authorized administrator.

- **Identification and Authentication**
  – The product/system shall provide functions for initializing and modifying subscriber authentication data.
  – The product/system shall restrict the use of these functions on the subscriber authentication data for any subscriber to the authorized administrator.
  – The product/system shall protect authentication data that is stored in the product/system from unauthorized observation, modification, and destruction.
  – The product/system shall be able to terminate the subscriber session establishment process and disable the subscriber account after five unsuccessful authentication attempts until an authorized administrator enables the account.
  – The product/system shall authenticate any subscriber's claimed identity before performing any functions for the subscriber.

- **Access Control**
  – The product/system shall verify applicable authentication and integrity services for the received transactions as determined by the standard compliant protocol.
  – The product/system shall apply applicable authentication, integrity, and confidentiality services to all transactions, i.e., requests and responses, as determined by the standard compliant protocol.
  – The product/system shall release the keys only to authorized subscribers.
  – The KRA shall release the key only if the requester is authorized to receive the key associated with the subscriber specified in the request and for the validity period (time) if specified in the request.
  – The product/system shall ensure that security policy enforcement functions are invoked and succeed before any security-related operation is allowed to proceed.
  – The product/system shall restrict the ability to perform security-relevant administrative functions to a security administrative role that has a specific set of authorized functions and responsibilities.
  – The set of security-relevant administrative functions shall include all functions necessary to install, configure, and manage the product/system. Minimally, this set shall include assignment/deletion of authorized subscribers from security administrative roles; association of security-relevant administrative commands with security administrative roles; assignment/deletion of subjects whose keys are held;

assignment/deletion of parties who may be provided the keys, product/system cryptographic key management, actions on the audit log, audit profile management, and changes to the system configuration.
  - The product/system shall allow only specifically authorized subscribers to assume only those security administrative roles for which they have been authorized.
  - The product/system shall define a set of security administrative roles that minimally includes security administrator, system operator, cryptographic officer, and audit administrator.

- **Nonrepudiation**
  - The KRA shall be able to generate evidence of receipt for received transactions.
  - The KRA shall be able to generate evidence of receipt of registration or deposit of KRI from subscribers.
  - The KRA shall be able to generate evidence of receipt of requests from requester.
  - The product/system shall generate evidence of origin for transmitted key recovery requests or responses.
  - The product/system shall provide a capability to verify the evidence of origin of information to the recipient.
  - The product/system shall provide a capability to verify the evidence of receipt of proof of receipt to the originator of message, i.e., recipient of proof of receipt.
  - The product/system shall provide the originator the ability to request evidence of receipt on information.

Availability

The KRA shall provide a secure replication of any KRI stored.

Protection of Trusted Security Functions

- The product/system shall provide a communication path between itself and local human subscribers that is logically distinct from other communication paths and provides assured identification of its endpoints.
- The local human subscribers shall have the ability to initiate communication via the trusted path.
- The product/system shall require the use of the trusted path for initial subscriber authentication.
- The product/system shall provide the authorized administrator with the capability to demonstrate the correct operation of the security-relevant functions provided by the underlying abstract machine.
- The product/system shall preserve a secure state when abstract machine tests fail.

Policy

The KRA shall have a written policy based on the KRA Policy Framework. It shall operate in accordance with this policy.

Registration Agent

Agents should protect all sensitive information from modification.

- **Nonrepudiation**—The RA shall be able to generate evidence of receipt for received transactions.
- **Integrity**—The RA shall be able to provide proof that information maintained has not been altered.

Licensing Agent

Licensing Agents shall perform compliance audit of the KRA to ensure that the KRA operates in accordance with the KRA's stated policy.

8.1.5.12 Administration

Having good policies and technical solutions will not ensure the secure operation of a KMI/PKI or the validity of the subscriber certificates. An extensive set of operational policies and practices supporting the technical solutions also has to be in place. Historically, many problems found with infrastructure have not been with the technology but with poor procedures; the operator did not know what to do in a given situation or the operator did not follow the proper procedures.

Administration of the infrastructure involves much more than the procedures to identify subscribers and create their certificates. It also requires managing the people, the components, and the networks making up the KMI/PKI. Because of the wide range of activities that impact the KMI/PKI's security, the administration function is spread across a large number of people. Each of them must do their jobs correctly to have the level of trust defined in the policy. Specific tasks include—

- Enforcing policy (e.g., compliance audits).
- Administrating the network and system elements.
- Managing the technical security mechanisms for the infrastructure elements (e.g., administrating the computer's access control list, reviewing the audit files).
- Performing key and certificate accounting.
- Managing the cross-certification process.

- Managing the compromise recovery process.
- Defining and documenting operational and security procedures.
- Training operators.
- Managing physical and personnel security.
- Providing disaster recovery mechanisms.
- Maintaining availability.
- Managing the key recovery process.

Establishing the KMI/PKI's trust relationship via a set of policies (Section 8.1.5.1, Policy Creation and Management) is the first step, but the trust model has to be continuously managed or it will become meaningless. The policies have to be translated into operational and security procedures for the specific technical solution employed within the KMI/PKI. These procedures provide a framework for the operators to administer the system. The procedures should cover all the normal processes in running the KMI/PKI and known exception and emergency activities. These procedures have to be documented and distributed to all appropriate KMI/PKI elements. The infrastructure should periodically reexamine and update the procedures as the policy changes, new processes are added, new exception cases are identified, new technical solutions are employed, or better ways of administrating the policy are found.

Once the infrastructure identifies and documents the procedures, the operators must be trained in the system policy and related procedures. Beyond the technical procedures necessary for their jobs, the operators must have an understanding of their responsibilities and limitations, and the security implications of not following the procedures. This process is open-ended because as the policy and procedures change, the operators need to be retrained.

The infrastructure has a responsibility to its subscribers and other domains to uphold its end of the trust relationship. This requires a mechanism to monitor the actions of every element in the KMI/PKI to ensure that they correctly implement the policy and procedures. Compliance audits, based on traditional concept of key management audits, are one way of tracking the subordinate elements. The root (or designated agent) periodically reviews each element to check the degree of compliance with the policy and procedures. The audit should also test little used and contingency procedures to determine if the operators would respond correctly. The results should identify and help correct problems with elements not properly implementing the procedures. Results should be available to other people in the trust relationship (e.g., domains that are cross-certified).

One of the most important extensions of the trust relationship is the addition of outside domains through a cross-certification. In effect, this gives every subscriber in the outside domain the same trust characteristics as an original member of the domain. This requires that the new domain have an equivalent level of assurance as the original domain (and vice versa). The only way to determine if this equivalency exists is to examine the two policies and determine whether they provide equivalent degrees of assurance. Standardizing on the format for documenting policies helps in comparing the policies by allowing a straightforward comparison of parallel certificate policy elements. One caution is that it is almost impossible to determine if the other domain actually implements their policy correctly (no independent compliance audit). The domains are trusting the correct enforcement of the other's domain security policy. Each domain has to monitor the other domain's performance and revoke the cross-certification link at any sign that it does not implement its policy correctly.

Because trust is the fundamental characteristic of the KMI/PKI, the physical and personnel security is important. A system operator can do extensive damage to the system and subscribers throughout the domain who rely on the certificates they authorize. Consequently, the people who have authorized access to the system must be trusted to do their jobs honestly, while all unauthorized subscribers are prevented from accessing the KMI/PKI.

Personnel security consists of both the hiring of the operators and their continued supervision. The owners of the infrastructure should perform some level of investigation (as defined in the policy) on their prospective employees to gain confidence in their trustworthiness. Periodic reinvestigations are necessary to maintain that degree of trust. If these reinvestigations or other actions bring their trustworthiness into question, those operators should be temporarily removed from access to the system. If further investigation confirms the suspicion, the keys and certificates they created may need to be revoked

Physical security provides for the isolation of the KMI/PKI elements from access by unauthorized people. Protection is required for both the physical elements and their relevant KMI/PKI information. The policy should define the level of protection required. Because of the different sensitivities of elements within the infrastructure, the protection may vary. For example, the root might be located in a no-lone, i.e., an area where two-person integrity is required, zone protected with a 24-hour guard while a low level CA might only need a lockable protective container.

The technical security requirements must also be managed. These include the computers and networks that are used to implement and transport the infrastructure and its products. While these do not provide subscriber services, they are susceptible to attacks. If corrupted, they can negate other security mechanisms in the system. The infrastructure needs the same set of services (e.g., computer security, network confidentiality, intrusion detection, as other applications), so many of the solutions defined in Chapter 5 are applicable to the KMI/PKI.

The system administrators for the network, firewalls, and computer systems have to ensure that the underlying equipment works and provides the necessary security. The system administration should be a unique role and not done by an operator. The network administrator is responsible for providing network security services, e.g., authentication, access control, availability, and protection from network attack, and setting up the firewall. The computer system administrator is responsible for providing computer security services (e.g., least privilege, review audit files, access control, and virus protection). They have to install the computer equipment, set up operator accounts, define operator access privileges, monitor operator activities, install new software, and install security software and patches. Administrator actions should be part of the compliance audit.

The KMI/PKI has to maintain its continuity in the face of an emergency that destroys infrastructure elements or during the routine elimination of existing infrastructure elements. That requires advance planning for each of the elements and the definition of appropriate disaster recovery mechanisms. Operators need to be trained in the recovery procedures, and they should be tested as part of the compliance audit. The disaster recovery plans should guarantee the availability of the following services and information:

- Ability for subscribers to access certificates and compromised information.
- Ability to generate and distribute compromise information.
- Ability for subscribers to verify existing certificates.
- Archived records.
- Key recovery information.
- Authenticated copies of the old system parameters, e.g., root public key.
- Ability to reconstitute KMI/PKI with existing elements by creating new root and adding new elements as appropriate.

8.1.5.13 Requirements

Requirements related to the operation of the KMI/PKI include the following:

- The KMI/PKI shall ensure that a key or certificate request comes from an authorized source.
- Before issuing a key or certificate, the infrastructure shall verify that all the information within the request is valid.
- The CA shall authenticate a subscriber requesting a certificate to ensure that the correct public key is bound to the proper identity.
- The CA shall notify a subscriber when it has generated a certificate for that subscriber.
- With the exception of special circumstances (e.g., revocation attributed to firing an employee), the CA shall notify a subscriber when it has revoked the subscriber's certificate.
- The KMI/PKI will notify all subscribers of a revocation of a symmetric key.
- The KMI/PKI shall provide timely key and certificate revocation information to its subscribers.
- CAs shall provide their public key and/or public key certificates to subscribers in a secure and authenticated manner.
- A CA shall protect the private key material that it uses to sign certificates.

- The CA shall only use its signing private key material to sign certificates.
- If the KMI/PKI generates either symmetric keys or asymmetric key material on behalf of a subscriber (e.g., traffic encryption key or key agreement key material), the infrastructure shall ensure that the material is generated securely and securely distributed to the subscriber.
- If the KMI/PKI stores subscriber private key material for recovery purposes, the infrastructure shall ensure that this information is protected in storage and is revealed only to the subscriber or to an authorized authority. It shall also ensure that the recovery key material is securely distributed to the subscriber or authorized authority.
- The KMI/PKI shall define a policy for the domain and ensure that all elements operate within the scope of that policy.
- The KMI/PKI shall account for the life cycle (ordering, generation, distribution, rekey, destruction and archive) of symmetric key and asymmetric key materials and certificates.
- Proper technical and procedural controls shall be implemented to protect the components of the KMI/PKI.

8.1.5.14 Attacks and Countermeasures

Attacks

Attacks that can be mounted against the KMI/PKI as a whole or to individual KMI/PKI components include the following:

- **Sabotage.** The KMI/PKI components or hardware token on which the subscribers or infrastructure elements keys and certificates are stored may be subjected to a number of sabotage attacks, including vandalism, theft, hardware modification, and insertion of malicious code. Most attacks are designed to cause denial of service. However, attacks such as hardware modification and insertion of malicious code may be used to obtain copies of subscriber or CA key material as they are generated, obtain information entered by the subscribers or operator such as a PIN, or cause known keys to be generated.
- **Communications Disruption/Modification.** Communications between the subscribers and the KMI/PKI components could be disrupted by an attacker. The disruption could cause denial of service, but may also be used by the attacker to mount additional attacks such as the impersonation of a subscriber or the insertion of bogus information, such as a key order, into the system.
- **Design and Implementation Flaws.** Flaws in the software or hardware on which the subscriber depends to generate and/or store key material and certificates can result in the malfunction of the software or hardware. These malfunctions may deny services. The flaws may accidentally or be intentionally exploited to disclose or modify keys or

certificates. Improper installation of the software or hardware may also result in similar consequences.

- **Operator Error.** Improper use of the KMI/PKI software or hardware by the operators may result in denial of service or the disclosure or modification of subscriber's keys and certificates.
- **Operator Impersonation.** It is possible that an attacker may impersonate a legitimate KMI/PKI operator. As an operator, the attacker would be able to do anything a legitimate operator could do such as generate key, issue certificates, revoke certificates, and modify other infrastructure data.
- **Corruption or Coercion of the KMI/PKI Operator.** It is also possible that a KMI/PKI operator might be corrupted or coerced by an attacker to generate unauthorized key, issue certificates to an unauthorized subscriber, revoke certificates of legitimate subscribers, and modify other infrastructure data.

Countermeasures

Countermeasures that may be implemented to protect the KMI/PKI and its components from the attacks outlined above include the following:

- **Physical Protection.** Physical protection of KMI/PKI component hardware, communications link with other infrastructure elements, and/or hardware tokens will counter many of the sabotage and communications disruption related attacks.
- **Good Design Practices.** Concerns over flaws in the software and/or hardware design may be alleviated if good design practices are followed during the development of the software and/or hardware used in conjunction with the KMI/PKI.
- **Testing.** Testing of the software and/or hardware may also be used to counter attacks to the system that result from the exploitation of flaws in the system.
- **Training.** Training of the KMI/PKI operators and administrators is vital to eliminating or at least reducing the possibility of inadvertent attacks as a result of subscriber error.
- **Strong Authentication.** Strong authentication of the subscriber by the KMI/PKI components greatly reduces the possibility of impersonation attacks.
- **Access Controls.** Software or hardware based access controls may be implemented at the KMI/PKI components to limit the possibility that an unauthorized attacker will gain access to the infrastructure software or hardware.
- **Encryption.** Encryption of the link between the subscriber and the KMI/PKI components reduces the possibility that an attacker may eavesdrop on the communications and try to disrupt or modify the communications.

- **Contingency Planning/System Backup.** —Backup of a subscriber's keys, certificates, and relevant software and hardware is the best mechanism for protecting against design flaws that result in system failure.
- **N-Person Controls.** Requiring multiperson control on sensitive PKI functions, such as the process of bringing a CA to an operational mode and the generation of CA key material, can limit coercion related attacks.
- **Auditing.** Auditing may not prevent attack, but it may be used to detect an attack and to identify the culprit. The presence of good auditing capabilities may also act as a deterrent to some attackers.
- **Personnel Selection and Screening.** Personnel chosen to perform KMI/PKI functions should be selected on the basis of loyalty and trustworthiness. People performing such functions should be adequately paid, and screened for a prior history, which would indicate a pattern of untrustworthiness.

8.1.6 KMI/PKI Assurance

Section 8.1.1, KMI/PKI Introduction, addressed the KMI/PKI as a menu with a set of independent processes with independent solutions. However, a KMI/PKI's security is actually based on the interaction among all the processes. Because the intelligent attacker will always attempt the easiest attack that meets their goals, it makes little sense to have processes at vastly different levels of security. The effect is only to drive up the development and operational costs without increasing the security posture. A better approach would be to determine the security level needed for each application supported by the infrastructure and to choose a set of solutions that correspond to that security level.

Providing a high-assurance KMI/PKI can be very expensive in terms of people and money. Cost effectiveness of many applications, like informal messaging, Web browsing, or those handling low amounts of money, are very sensitive to PKI costs. For these applications, the KMI/PKI cannot cost more than a fraction of the potential loss from a successful attack. These applications may be willing to settle for a KMI/PKI that provides low cost certificates, but does not have all of the procedural and technical protections in place against certain attacks. In effect, KMI/PKI security is a form of insurance and employs the same cost considerations. A $1 certificate is acceptable for protecting a $100 transaction, but a $50 certificate is not appropriate to protect the same $100 transaction. Other applications may be willing to pay the added cost for better procedural and technical protections because the certificate is protecting more valuable information. A $50 certificate might be acceptable if it is protecting a $100,000 transaction.

There is much ongoing work in the standards community and the Government in grouping the individual process solutions into fully developed architectures with common security standards. Among the groups working to define these standards are the IETF, FPKI, DoD, Canadian government, and commercial certificate providers.

8.1.7 KMI/PKI Solutions

Examples of KMI/PKI usage will illustrate the practical aspects of system design. Three categories of systems—DoD, Government, and corporate, each with important design and functional characteristics—are presented. Each example of the KMIs is actively involved in upgrading its information assurance (IA) assets and applications. The first category presented begins with summaries of the DoD Class 3 PKI and the FORTEZZA PKI followed by a detailed description of the target DoD KMI/PKI system showing its architectural development concerns, considerations, and issues. The lengthy description typifies the broad aspects of planning and considerations associated with a secure infrastructure implementation plus the added protections needed for processing classified information. This example demonstrates the challenge of designing a large system in today's environment. The DoD anticipates continued growth in the demand for security support for classified applications and Class 3 and Class 4 PKI capabilities. The Government KMI/PKI Solutions category will be presented next to show the many similarities with the DoD KMI/PKI despite its emphasis on UNCLASSIFIED//FOR OFFICIAL USE ONLY (U//FOUO) information. An example from the U.S. Government will be presented. The Federal KMI/PKI description includes the concept of bridging trust paths among PKI communities. The Corporate Solutions category is filled with a myriad of commercial-off-the-shelf (COTS) products and services. Several are presented followed by a summary sketch of a corporate system. A short description using Kerberos for KMI security is provided to show some of the work being performed in academia.

8.1.7.1 DoD Class 3 PKI

The following summary highlights the DoD Class 3 PKI.

PKI Name

The PKI name is the Department of Defense Class 3 Public Key Infrastructure (DoD Class 3 PKI). The original term, "medium assurance," may be used interchangeably with the term, "Class 3."

The following summary highlights the Department of Defense Class 3 Public Key Infrastructure (DoD Class 3 PKI) Solution, a forerunner of the DoD Target KMI/PKI described later.

PKI Design and Operational Responsibility

The overall program management of all DoD efforts required to meet the goals and milestones in the DoD PKI Roadmap is the responsibility of the DoD PKI Program Management Office (PMO). The National Security Agency (NSA) is the PMO Program Manager with the Defense Information Systems Agency (DISA) providing the Deputy Program Manager leadership.

NSA is responsible for defining the security architecture and security criteria for the DoD PKI. This includes criteria for the components and their operation. NSA (or an approved National

Information Assurance Partnership [NIAP] vendor) will evaluate the security of products and services employed in the DoD PKI. Figure 8.1-14 illustrates the architecture of the current DoD Class 3 PKI, minus the Root CA. The Class 3 PKI Root CA is located at the NSA Central Facility.

PKI Subscriber Community and Applicability

One element of the Defense-in-Depth strategy is the use of a common, integrated DoD PKI to enable security services at multiple levels of assurance. The DoD PKI provides a solid foundation for IA capabilities and general-purpose PKI services (e.g., issuance and management of CRLs in support of digital signature and encryption services) to a broad range of applications, at levels of assurance consistent with operational imperatives.

Classes 3 and 4 have been defined to support the protection of nonclassified mission critical, mission support, administrative, or format sensitive information on open networks (i.e., unencrypted networks). These PKI classes also can be used on closed networks (i.e., encrypted system-high networks such as Secret Internet Protocol Router Network [(SIPRNet]) to provide additional protection such as subscriber authentication and data separation/communities of interest (COI). Specifically, Class 3 certificates and applications are appropriate for many business transactions, in which the monetary value of the transaction or the sensitive or unclassified information protection is moderately high. By contrast, the Class 4 PKI products and services will be used to protect sensitive or unclassified mission critical information in a high-risk environment such as the Nonclassified Internet Protocol Router Network (NIPRNet). In addition, the Class 5 PKI products and services (still in the planning stages) will be used for the protection of classified information on open networks or in other environments in which the risk is considered high.

PKI Products

The DoD PKI uses COTS products to keep up with technology evolution and develops government off-the-shelf (GOTS) solutions when necessary. The newness of standards and products, however, may cause some interoperability problems among vendors' products. The DISA and NSA actively work with vendors and standards communities to develop standard specifications and implementations that improve interoperability. The DoD is committed to ensuring that DoD specifications are consistent with the emerging commercial and National Institute of Science and Technology (NIST) federal standards to support DoD interoperability requirements.

PKI Future Plans and Schedule

The majority of activity to date in the DoD PKI arena has focused on understanding the technology, the standards, operational policy and procedural issues and on establishing the role of PKI relative to the remainder of the IA Defense-in-Depth model. The experiences gained from the two major DoD PKI initiatives the development and deployment of an operational FORTEZZA PKI, in support of the DMS and other FORTEZZA-enabled applications, and the pilot medium assurance PKI—have been instrumental in the development of the target DoD PKI architecture.

The DoD PKI will initially support three levels of assurance, defined as Classes 3 and 4 (formerly, Medium and High) for the protection of unclassified/sensitive information, and Class 5 (for the protection of classified information on unencrypted networks). The long-term goal is to provide a Class 4 certificate to all DoD personnel and—where appropriate—Class 5 certificates via the target DoD PKI. Each assurance level has its own set of requirements for technical implementation and process controls, which becomes more rigorous as the level increases.

The target DoD PKI shall employ centralized certificate management and decentralized registration and shall use common processes and components to minimize the investment and manpower to manage and operate the PKI. The target DoD PKI also shall support a broad range of commercially based, security-enabled applications and shall provide secure interoperability with the DoD and its federal, allied, and commercial partners while minimizing overhead to and impact on operations.

The DoD PKI program continuously tracks new and evolving IETF standards to ensure that the most viable commercial standards are fully leveraged to support maximum interoperability in the future.

In addition, to ensure secure interoperability between DoD and its vendors and contractors, External Certificate Authorities (ECA) will be established using a process that ensures the required level of assurance to meet business and legal requirements. ECAs will be approved by the DoD Chief Information Officer (CIO), in coordination with the DoD Comptroller and the Office of the Secretary of Defense (OSD) General Counsel.

The DoD PKI will be implemented in a series of actions to reach the final goals. These actions are as follows:

- All DoD organizations must deploy registration applications for supporting the Class 3 (formerly Medium Assurance) PKI and the Class 4 (FORTEZZA-based) PKI.
- Protection of Category 1 mission-critical systems on unencrypted networks using Class 4 certificates and tokens.
- Protection of Category 2/3 mission-critical systems operating on unencrypted networks must use Class 3 certificates.
- Protection of Category 2/3 mission-critical systems operating on unencrypted networks must use Class 4 certificates and tokens.
- Server Authentication.
- Client identification (ID) and authentication.
- Private DoD Web servers access control software for Class 3 certificates.

- E-mail applications to facilitate digital signature processing of all individual messaging within DoD using Class 3 certificates.
- ID card processing software, building and facility access software, and workstation access software applications shall begin implementation for Class 4 certificates.

Additional Information

The following sources have additional information on DoD Class 3 PKI products and services and the DoD PKI:

- Requesting Use of the DoD Pilot Medium Assurance Component of the DoD PKI (explains information and feedback to be provided to use the DoD Medium Assurance Pilot)
- DoD Medium Assurance Public Key Infrastructure (PKI) Home Page: http://ds-2-ent.den.disa.mil/
- U.S. DoD X.509 Certificate Policy, version 5.0,13 December 1999; and DoD PKI Roadmap, Version3.0, 29 October 1999.

8.1.7.2 FORTEZZA© PKI

The following summary highlights the FORTEZZA© PKI Solution, a solution being used by the DMS.

PKI Name

The PKI name in the FORTEZZA© CMI. A CMI differs from a PKI because it includes only the CMI and the policy associated with the CMI, not the directories where the public data items are posted.

PKI Design and Operational Responsibility

The KMI Services and Workstation Technology division (NSA) is the Certification Authority Workstation (CAW) PMO responsible for its design, development, and testing. The Requirements and System Engineering division (NSA) and the Life-Cycle Engineering and Standards division (NSA) are responsible for CAW life-cycle support issues, such as training, installation, upgrades, and maintenance. The Electronic Key Management System Operations division (NSA) is responsible for the FORTEZZA CMI operations. Actual CAW training is accomplished via a combination of classroom, computer-based, hands-on, and on-the-job training, per policy, with the classroom training conducted by General Dynamics (CAW 3.1), Motorola (CAW 4.2.1), and Service Schools (both CAW 3.1 and 4.2.1).

Figure 8.1-15 illustrates the Policy Approving Authority (PAA), Policy Creation Authority (PCA), Indirect Certificate Revocation List Authority (ICRLA), CA, RA, and Certificate Management User Agent (CMUA). Other CMI roles not requiring dedicated workstations are the System Administrator (SA) and the Information System Security Officer (ISSO).

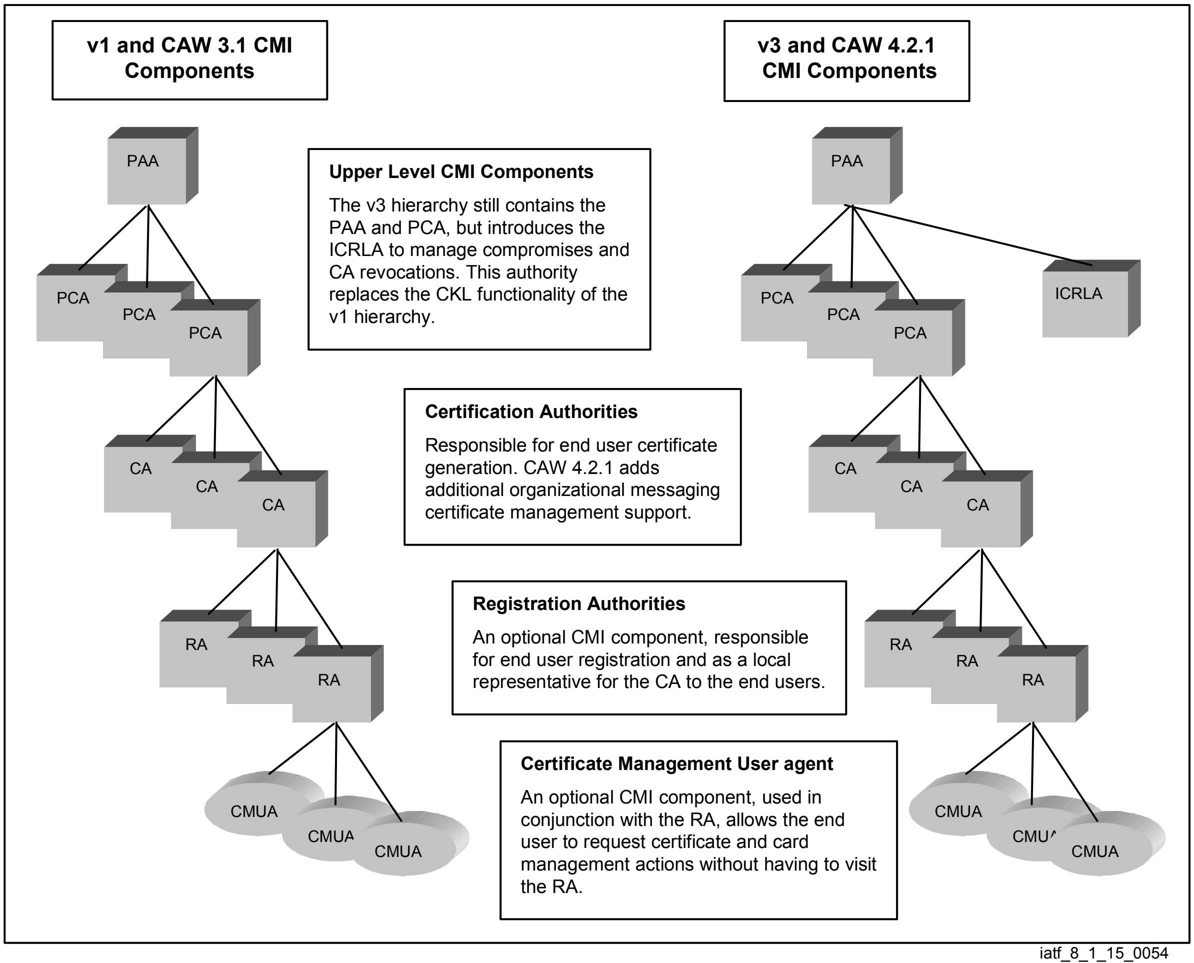

Figure 8.1-15. FORTEZZA CMI Components

PKI Subscriber Community and Applicability

The FORTEZZA PKI was targeted and is established to address certificate and security requirements of the DoD community, but its design and capabilities are also flexible to support civilian and commercial subscribers.

For DoD subscribers, the FORTEZZA PKI operates in compliance with Class 4 assurance policy resulting from its software design/development compliance with Trusted Software Design Methodology (TSDM) guidelines, its operation on a trusted operating system designed for the B1 level, implementation of high-grade cryptographic algorithms and keys, and its strict use of hardware tokens for system infrastructure components. The certificates created and managed by the FORTEZZA PKI, when teamed with compatible applications, enable subscribers to apply all of the security services—authentication and identification, confidentiality, privacy or data integrity, nonrepudiation, and access control—to unclassified and classified data. In addition, because of the high-grade cryptographic algorithms, keys, and tokens that the FORTEZZA PKI implements, it is possible for applications to provide protection (authentication and confidentiality) for information to cross-classification boundaries when such a crossing is already permitted under a system security policy (e.g., sending unclassified information through a High-Assurance Guard (HAG) from SIPRNet to NIPRNet).

DoD organizations and customers of the FORTEZZA PKI can operate CAs in their local, decentralized environment and are responsible for complying with either NAG-69C, Information Systems Security Policy and Certification Practice Statement for Certification Authorities (for X.509 vl use with CAW 3. 1 and CAW 4.2.1) or DoD Certificate Policy, Version 5.0 (for X.509 v3 use with CAW 4.2.1).

PKI Products

The FORTEZZA PKI supports secure DoD transactions across existing national and global information networks (e.g., Internet) and allows them to be protected from threats from other subscribers of the global information network. The functionality of the current FORTEZZA PKI, based on CAW 3.1, supports the following:

- Supports U//FOUO and classified environments on the same CA platform.
- Performs trusted downgrade of information between different classification levels of network(s)/account(s).
- Creates and manages X.509 vl certificates.
- Creates and manages vl CRL.
- Creates and manages card, certificate, and DN in a flat file database.
- Manually posts certificates, CRLs, CKLs to X.500 DSA.
- Processes MISSI Management Protocol (MMP) messages from other networked devices.
- Implements Message Security Protocol (MSP) 3.0.
- Manages backup data for certificates, CRLS, and CKLs.

The CAW 3.1 can be configured to serve as a PAA, PCA, or CA.

The optional RA using the Motorola Registrar product, provides a cost-effective alternative to dedicated CAWs for multiple subscriber registration and routine subscriber certificate update tasks. Registrar 4.2 is available now to support not only CAW 3.1 but also CAW 4.2.1 when it is fielded.

The optional Motorola CMUA resides on subscriber Windows NT platforms and further off loads subscriber registration and maintenance functions from the Registrar. This product is available now to support CAW 3.1 and CAW 4.2.1 when it is fielded.

PKI Future Plans and Schedule

The FORTEZZA PKI (X.509 vl certificates only) has been operational since March 1995. The current PKI (operational since January 1998) is based on CAW 3.1. An upgrade to CAW 4.2.1 began in March 2000 for the PAA and PCAs and staggered upgrades of the CAs in the field. The March 2000 upgrade is backward compatible to CAW 3.1 functionality and its X.509 vl certificates. The March 2000 upgrade also provides a totally new software design and code based on TSDM Level 3 guidelines, a new and improved GUI, a relational database, automatic posting of information to a public directory, management of multiple hardware and software tokens, programmable X.509 certificate extensions for flexible security policies, X.509 v3 certificates, v2 CRLS, and Indirect Certificate Revocation Lists (ICRL).

Plans are under way to develop and field a future CAW version to provide support for software FORTEZZA technology and capabilities.

Additional Information

Additional information can be found in the following documents:

- Interim Operational Security Doctrine for the Unclassified but Controlled FORTEZZA Card, 18 February 1998.
- Interim Operational Security Doctrine for the FORTEZZA for Classified (FFC) FORTEZZA Card, June 1998.
- NAG69B, Information Systems Security Policy and Certification Practice Statement for Certification Authorities, 24 October 1997 (for X.509 vI with CAW 3.1 and 4.2.1).
- NAG69C (replacement for NAG69B, pending final approval at NSA)(for X.509 vl with CAW 3.1 and 4.2.1).
- DoD Certificate Policy, Version 5.0 (for X.509 v3 with CAW 4.2.1).
- FORTEZZA Public Key Infrastructure (PKI) Concept of Operations (CONOPS), Version 1.8, 7 January 2000.
- Certificate Management Infrastructure (CMI) Transition Plan, Version 2.0, 23 November 1999.

8.1.7.3 DoD Target Key Management Infrastructure

Throughout the following text, KMI is used interchangeably with KMI/PKI.

8.1.7.3.1 Background

The people, programs, and systems that carry out or support the broad range of DoD missions perform a variety of activities. These diverse activities, depicted in Figure 8.1-16, represent an ever-expanding need and role for IA capabilities in DoD operations. Traditionally, DoD has addressed these needs with stand-alone cryptographic components. In today's IT-rich environment, DoD's IA needs are being addressed with security features integrated into the many communications and information processing system components used by the DoD. These include workstations, guards, firewalls, routers, in-line network encryptors, software applications, and trusted database servers. The deployment of the large numbers of these security-enabled components (both traditional cryptographic devices and integrated IA features) is placing an increasing burden on the network infrastructure that provides KMI products and services.

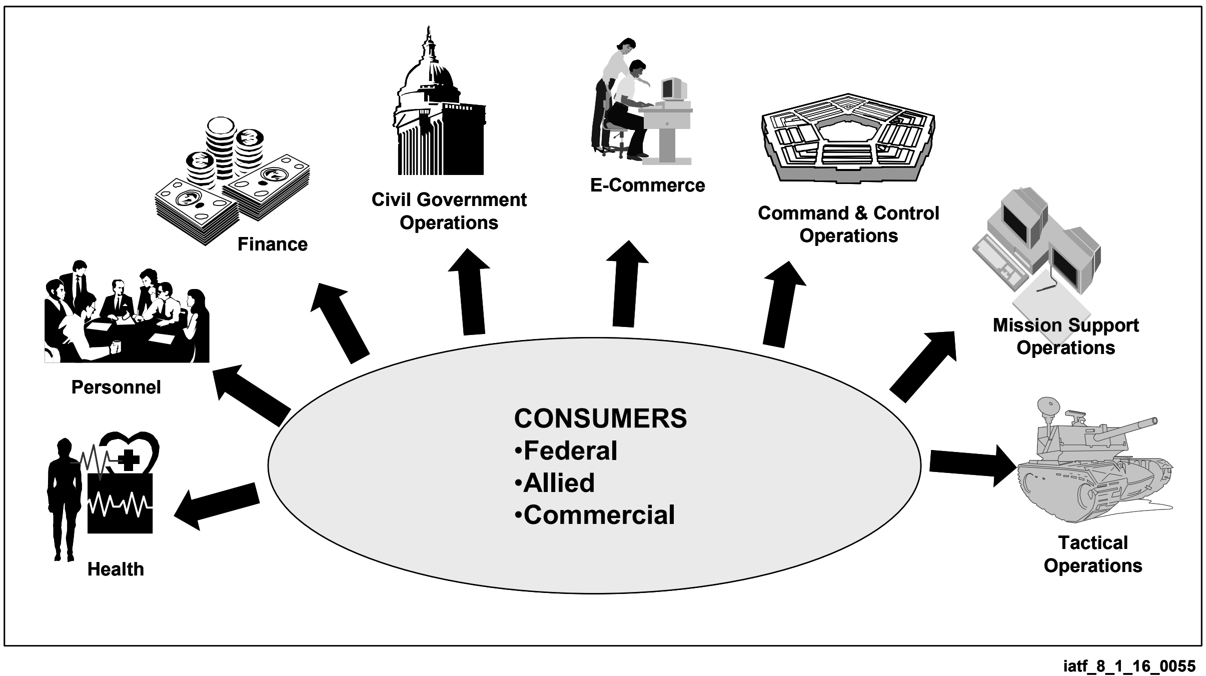

Figure 8.1-16. Operational Activities Supported by the KMI

The DoD KMI is a foundational element for a secure IA posture in the Defense Information Infrastructure (DII) and the broader national security community. The DoD is taking an aggressive approach in acquiring a KMI that meets the requirements for all IA key management needs. The DoD KMI program, supported by the services and agencies, Joint Staff, and DoD contractor community, is addressing this critical need.

The state of the current key management systems creates compelling reasons for modernizing the DoD KMI.

- **Infrastructure of Independent Stovepipes.** The current key management environment is composed of separate and independent infrastructures that provide and manage their own set of security products. These systems will become increasingly cumbersome and costly as new technology and their attendant security solutions continue to advance and the resources needed to operate them decline. This key management environment is composed of several unique solutions built for specific product lines. Although the solutions satisfy unique security needs, they each require different tools and training to obtain their respective products and services, imposing an unwarranted strain on resources.

- **Inefficient Expansion of New Capabilities.** Adding new key management capabilities has frequently required integrating new capabilities into existing systems that were not designed to perform the new functions, or creating new, independent systems to provide the needed support. One recent example is the deployment of a totally separate (stovepipe) network infrastructure to support DoD's use of PKI-based security products. Although this is an example of the limitations of the existing KMI structure, other programs are running into the same issues. This is impeding DoD's ability to respond to new requirements and demanding more resources for supplying cryptographic key products to support its missions.
- **Common Functions and Operations.** Although created independently, the existing systems contain many common threads (e.g., registration, ordering, and distribution) that could logically be combined and offered as a unified set of processes. The key management community and DoD Joint Staff have recognized this fact. They have identified a unified KMI as a critical system infrastructure that is needed to support key and certificate management approaches for mission-critical, logistic, and administrative systems.
- **Opportunities for Applying New Technologies.** Several KMI systems that have existed for a number of years are in need of an upgrade to take advantage of modern communication technology. This technology area has advanced significantly in recent years, providing the marketplace with many new and worthwhile, applicable techniques that would greatly improve efficiency and performance.

Given the critical importance of key management, applying modern technology within a sound IA systems approach is imperative. The KMI initiative focuses on unifying the disparate key management systems within a single, modern architecture—one that is modular, flexible, and extensible and will eliminate redundant resources associated with operation, maintenance, and training, resulting in substantial cost savings.

Commercial security technology using public key cryptography for U//FOUO requirements is rapidly becoming the largest "application class" that must be supported by the DoD KMI. However, requirements for support of classified applications are also projected to continue to grow significantly as new classified solutions such as secure wireless and Global Positioning System modernization are implemented. This creates the need for a more encompassing key management paradigm. The KMI will enhance the DoD's capability to support these mission-critical requirements. The DoD KMI program will unify these many disparate key management systems within a single, modern framework, introduce additional key management capabilities to support the continued expansion of KMI services that are projected, and address the Congressional mandate to reduce operational costs associated with the KMI.

KMI Products and Services

KMI, as described herein, refers to the framework and services that provide registration, enrollment, generation, production, distribution, control, and tracking of the broad range of KMI products needed by the DoD. A critical challenge for the KMI will be to provide continuing support for existing products and services and for emerging security solutions. At a minimum, the following product categories will be supported:

- Human-readable cryptographic products (e.g., code books, one-time pads, authenticators, and key lists).
- Symmetric cryptographic key for point-to-point and net use and for use in wireless products.
- DoD Class 3 PKI Root CA.
- Asymmetric cryptographic products.
- Electronic certificates (e.g., signature, attribute, and key exchange) used in a multitude of applications to implement security functions such as I&A, access control, integrity, confidentiality, and nonrepudiation.
- Key management documentation (e.g., policy documents, equipment operator manuals, and specifications) needed in support of the cryptographic user community.

The Target KMI provides the framework and services that unify the secure creation, distribution, and management of these products. The DoD KMI will enable the provisioning of these services for military, intelligence, allied government, contractor, and business customers. A baseline set of key management services offered by the KMI to support the user community includes the following:

- **Registration**—Identifying, in an authenticated manner, individuals, or system entities (either internal or external to the KMI) and their related attributes.
- **Enrollment**—Authenticating the establishment, modification, and deletion of privileges for individuals, system entities, or organizations.
- **Ordering**—Requesting cryptographic product (e.g., keying material, certificates, and manuals) to support a security application.
- **Generation**—Generating cryptographic products (e.g., symmetric key, asymmetric key and/or a public key certificate) by a security infrastructure element.
- **Distribution**—Providing physical and electronic products, including rekey, to the user in a secure, authenticated manner.

- **Policy Management**—Managing and enforcing policy and procedures for operating the KMI in a trusted and secure manner.
- **Trust Extension**—Reviewing and ruling on issues of cross-certification or bridging with other key management infrastructures.
- **Archiving**—Providing for long-time storage and retrieval of important data that may not be immediately accessible to online users of the system.
- **Accounting**—Tracking the location and status of cryptographic products.
- **Key Recovery**—Recovering encrypted information when the intended decryption key is unavailable.
- **Compromise Management**—Providing notification of compromised keys and invalid certificates in a timely and authenticated manner.
- **Audit**—Supporting periodic security evaluation of KMI operations.
- **Library**—Providing access to key management reference documents and information.
- **Destruction**—Destroying certificates and keying material.

Planned Evolution

The DoD KMI will be implemented as a series of evolutionary phases culminating in a re-designed, unified architecture. Strategic and architecture planning will require indepth coordination with KMI government and commercial partners. Every 18 to 24 months, a new Capability Increment (CI) will be delivered to operational users taking into account new and updated user operational, security, policy, and technology requirements, and programmatic opportunities. Timing of the capability increments is critical to ensure optimum synergy and cohesion with the individual systems in the DoD KMI architecture. For each CI, the Target KMI will be redefined to be consistent with current and projected operational/security needs and technology advances. The updated Target KMI definition will be used for programming and budget planning for the products and services needed to realize the Target KMI. This approach requires sustaining system engineering and development resources, and wide service/agency/ organization support for the acquisition, deployment, and operations of each CI.

The KMI uses a wide variety of existing networks and workstations to fulfill its mission and is being designed to implement as many KMI-wide functions as possible on COTS platforms. Initial deployments of the KMI will be structured as separate KMI functions for each security classification domain. However, as the system evolves, it will transition to a structure that allows the transfer of appropriate data between domains. Using this approach, most KMI functions will operate on a single-level (commercial) system-high platform at client manager nodes and in the centralized portions of the system infrastructure.

Goals and Objectives

A number of goals have been identified for the KMI based on user community input security, advancing technology and the reality of a shrinking budget. These goals are as follows:

- **Transparency.** Although some functions within the KMI inherently require direct operator or user interaction, the KMI will automate as many operations as possible. KMI-aware devices will interact with the KMI, transparent to the user.[1] Current, manpower-intensive operations (including accounting and archiving) will be automated and transparent to KMI users.
- **Ease of Operation.** The Target KMI will provide simplified, intuitive, and consistent interfaces for users to obtain KMI support for the ever-increasing range of PK functions. Users will have standard Web browser access to the KMI—with screens tailored based on their identity, role (and authorized capabilities), and KMI products and services tightly integrated into their mission planning and system management capabilities.[2]
- **Access to Needed Information.** The KMI will offer direct, online access on all relevant policy information and to operational information (e.g., inventories of keying materials and cryptographic devices) to ensure that policies are carried out appropriately. Customer support will be provided 24 hours a day, 7 days a week to assist users with KMI-related issues.
- **Reduction in User Manpower Support Needs.** Continued proliferation of cryptographic devices (user terminals, network servers, security-enabled network devices) and projected wide-scale deployments of PKI-enabled software applications will continue to increase user manpower burdens to obtain KMI products. The Target KMI will reduce this burden with its greater use of commercial standards and products.
- **Responsive Policies and Doctrine.** Uniform, national level, and DoD-wide policies, doctrine, practices, and procedures will be established in joint-community forums to ensure interoperability and consistency of joint operations at the organizational level. They will be coordinated and issued before deployment of cryptographic equipment.
- **More Efficient Use of KMI Operator (Internal) Support Needs.** Continued proliferation of cryptographic devices (user terminals, network servers, security-enabled network devices) and projected wide-scale deployments of PKI-enabled software applications will continue to increase demands on KMI operator manpower needed to generate and produce KMI products. The Target KMI will be more efficient than the existing KMI, allowing them to deliver products faster and respond more quickly to new requirements.
- **Enhanced Security.** Delivery of all orders will be available securely and directly to the end-user or end-user devices that require them. The KMI will be built on authentic,

---

1 Although the KMI can provide secure infrastructure capabilities to enable this transparency, modifications to KMI-aware devices are also required to add functionality that can realize this transparency.

2 Similarly, this goal can be realized only with enhancements to mission planning and system management components.

universally accepted identities for all users, operators, and devices. Standard tools and tool kits will be provided by the KMI to ensure that all KMI-relevant operations (e.g., key exchange, rekeying, and certificate path validation) are performed correctly.

General Features of the Target KMI

Several pervasive characteristics of the Target KMI exist:

- **Modularity.** The Target KMI, although still being refined, is based on a modular structure that will enable adequate flexibility to ensure that it can evolve over time. It will immediately leverage existing key management system capabilities and commercial components (e.g., commercial certificate authority workstations, directory systems) in the baseline implementation and incrementally evolve the capability as commercial technology matures. The KMI capabilities will evolve, taking advantage of commercial technologies; a strategy that requires a DoD enterprisewide standards approach, and a coordinated process within DoD to influence the direction of commercial standards bodies to incorporate features important to the DoD.

- **Automated Service.** The KMI will offer a well-defined set of KMI products and services, with an established set of delivery mechanisms and interface standards for "last-mile delivery devices," clearly defining how KMI products will be delivered.

- **Key Delivered Directly to End-Devices.** The KMI will evolve toward the electronic delivery of key, with delivery directly to end-devices. The KMI will provide tool kits that can be used to KMI-enable devices and operational support systems to take full advantage of the advanced features and capabilities that the KMI will offer.

- **Common Management Functions.** The KMI will introduce a set of common management functions that will enable consistent KMI operations provided by the various existing stovepipe KMI systems. It will augment these with a set of primary services (e.g., registration, common ordering, and key recovery) that will enable common KMI interactions for users and KMI-aware devices to obtain the specific KMI products or services they require. It will also incorporate functional and physical modularity to facilitate an orderly introduction and enhancement of operational capabilities throughout the KMI's life cycle.

- **Online Customer Support and Library Access.** The KMI will include an online repository to provide authorized KMI users and managers with a complete catalog of KMI products and services, test results of commercial IA products, electronic versions of current policies, manuals, advisories, and inventory status for deployed KMI-relevant devices and KMI products (including those of allies and coalition partners).

- **Leveraging Existing KMI System Investments.** The KMI encompasses products and services provided by the Electronic Key Management System (EKMS) physical key management capabilities and operational PKI capabilities. These provide a wide range of cryptographic keys for traditional symmetric key systems and key pairs and certificates for public key systems. The Target KMI provides the framework and services that will allow DoD to incorporate the existing KMI systems into the Target, thus improving the existing underlying system infrastructure that provides security services to military, intelligence, allied government, contractor, and business customers.

- **National Level Policies.** DoD faces many KMI challenges. It is anticipated that the implementation of DoD KMI will result in changes to areas such as national cryptographic policy to better coordinate the handling of classified and nonclassified key management data.

System Context

The KMI interacts with numerous external components and systems to perform its intended functions. Figure 8.1-17 illustrates the KMI system capability. A primary capability is to interact with the users it is intended to serve. The KMI must also interact with external federal and commercial KMIs and PKIs. It interfaces to external data sources, including local user community repositories and external data sources such as the Defense Eligibility and Enrollment Reporting System (DEERS) database. The DEERS database contains personnel information that may be accessed during registration of some end users.

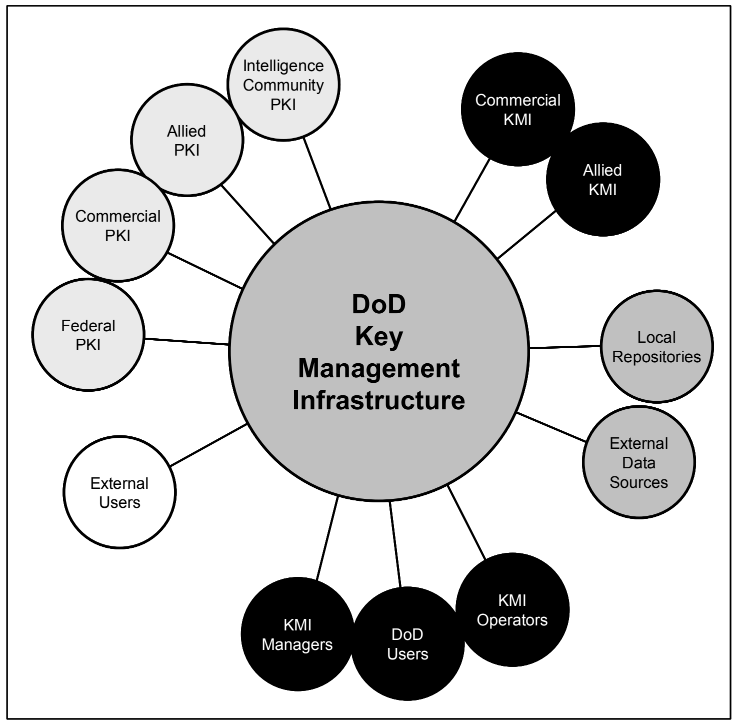

Figure 8.1-17. DoD KMI System Context

8.1.7.3.2 DoD KMI System Context KMI Nodal Architecture

The Target KMI architecture consists of four types of functional nodes, as shown in Figure 8.1-18. Their interconnectivity and summary of the major functions of each node is included in the figure, and discussed in detail below. Section 8.1.7.7 identifies the major documents that describe the Target KMI in detail.

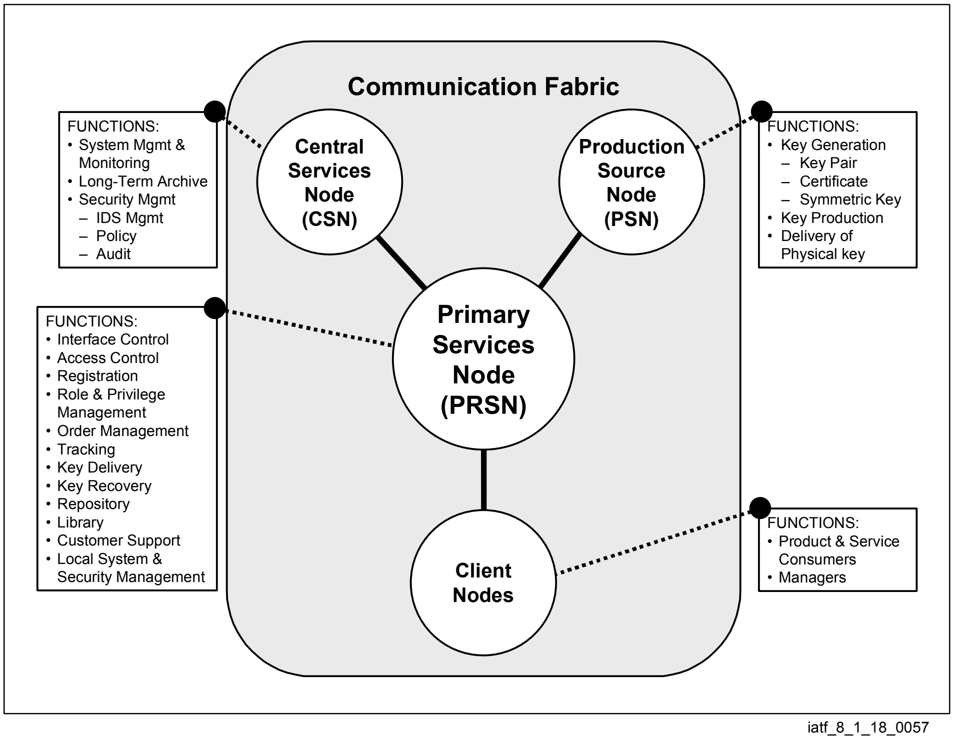

Figure 8.1-18. Nodal View of the Target KMI

Client Nodes

The client nodes represent the consumers of KMI/PKI products and services and the workstations that support the various KMI/PKI managers. Figure 8.1-19 provides a breakdown of several generic types of clients. Client nodes, also referred to as end entities, include stand-alone cryptographic devices, devices that incorporate security features that rely on key management services (e.g., security features within a router), and workstations that use software applications that require KMI support.

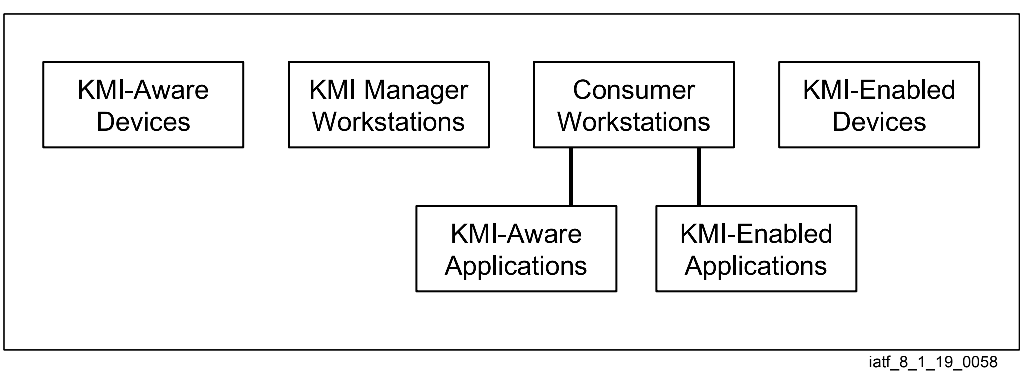

Figure 8.1-19. Breakdown of Client Nodes

Primary Services Node

KMI users—whether they are humans, devices, or applications—obtain their products and services from a Primary Services Node (PRSN). The PRSN provides common management functions in a server-based architecture and provides its required services in multiple classification domains. The PRSN provides to the client node components, unified and transparent access to all of the different production sources and delivery of KMI products and services to consuming applications, directly or through an intermediary. As implied in Figure 8.1-19, the PRSN is also the node that handles user access. When KMI products are requested, the PRSN will forward the request to the appropriate Production Source Node (PSN) for generation and production. If the product can be delivered electronically, the PRSN will forward it on to the client node.

Production Source Node

The PSN are responsible for the generation and production of KMI products. These products will be created at the request of PRSNs. If a physical product is needed, the PSN is responsible for delivering the product directly to the client node. PSNs are separated from the common management functions of the PRSN, but interface via available communications networks to the management infrastructure provided by the PRSN. The EKMS Central Facility and Key Processor (KP), the existing physical systems, and the PKI CA are examples of current KMI systems that provide functionality associated with a PSN. The Target KMI architecture has adopted a modular structure specifically to accommodate the modification of existing, or addition of new production sources.

Central Services Node

The Central Services Node (CSN) provides overall system management and monitoring functions for the system infrastructure. In the Target KMI, the CSN will provide the long-term system archive and the master KMI database, and will replicate data to the individual security enclaves of the PRSNs. The CSN will also handle overall system infrastructure security management, including IDS oversight, audit data collection and analysis, long-term archiving, policy management, and system health monitoring.

General Deployment Considerations

The Target KMI will be deployed as modular sites consistent with the nodal architecture discussed above. There will be one CSN and a physically isolated hot backup to mitigate risks of natural disasters interrupting operation. Several PRSN will be sites in strategic locations across the Continental United States (CONUS).

Each will be capable of serving as a backup capability to other PRSNs, with automated cutover capabilities available to ensure uninterrupted service to KMI clients. Deployable versions of PRSNs will be established in sites outside CONUS to minimize network connectivity issues for operations in various theaters. Typically, these sites will reach back to the CSN and PSN located in CONUS. To the extent that PSN capabilities are needed to support these deployed sites, a black PSN will be available to provide the capability (using stored materials that can be transferred via physically and/or electronically protected means), minimizing the risks of operating in potentially hostile environments. The deployed PRSNs will also include basic CSN provisions to facilitate operations when connectivity back to CONUS is impaired or unavailable.

The KMI will use the communication channels already serving its customers in other capacities. The KMI will rely on existing communications paths for connectivity within the system. The KMI will also support dialing capability through secure terminal equipment. Once connections are established, the interfaces and functionality will be the same as that available when connecting to the KMI through a data network.

8.1.7.3.3 Perspectives on KMI Operations—An External KMI Perspective

This section provides an operational overview of major Target KMI functions from the perspective of users and managers of KMI products and services. Further detailed descriptions of these operations can be found in the Target KMI Concept of Operations Document.

The Target KMI is designed to automate operations to the extent that it is feasible and prudent. For those operations requiring human intervention, the KMI provides standard operating procedures for a range of user and manager functions (referred to as common management functions). KMI user and manager operations will be performed as local client workstations interacting with server capabilities in the PRSN. In general, a KMI user or manager will insert their KMI token into their workstation, log into the PRSN, request a particular KMI function, and be connected to the appropriate server.[3] Where feasible, the PRSN will provide intuitive screens with pull down menus tailored to the specific role(s) and privileges of the requester.

Registration

Registration is the process that allows an end entity to become *known* to the KMI. It establishes the identity of the end entity that the KMI asserts for all of its operations. Registration also results in the generation of an identity certificate and the creation of a token that is delivered to, and remains in the possession of, the registrant. KMI registration is a decentralized process that is performed by a number of Registration Managers (RM), including RAs and LRAs. Within the context of the Target KMI architecture, the RM is a client node manager and is typically someone who is located close to the user.

The DoD PKI Certificate Policy (CP) establishes the requirements and policies that are used during registration. In a typical scenario, a registrant appears in person before an RM and presents credentials of his or her identity as required by the appropriate CP and CPS. To register devices, the device sponsor or component administrator submits appropriate documentation about the device to their LRA. The RM logs into the PRSN using a KMI token to establish

3 Although reference is made to specific “server,” in actuality it represents functional capabilities of the KMI node referenced.

privileges and accesses the registration server. The RM validates that the information provided by the individual agrees with independent identity data obtained from an independent external data repository (e.g., DEERS database or a repository provided by the department, agency, or organization).

Enrollment

Enrollment is the association of privileges with an individual's KMI identity by a KMI Privilege Manager (PM). Enrollment enables KMI users and managers to conduct transactions for which they have been granted privileges. Each KMI operator and each client node manager has a defined role (or set of roles) in the KMI, and roles determine the scope of privileges within the security infrastructure. For example, the role of an RM, like an LRA, is to register users in the system. Other managers, such as user representatives or product requesters, may order keys, certificates, or other services from the KMI on behalf of registered users. A PM performs the function of defining roles, allocating privileges to those roles, and assigning roles to individual managers.

Request and Tracking

The process of requesting KMI products and services and then tracking the status of those requests is structured in a manner similar to registration and enrollment. Provisions are also included for direct requests to be made from KMI-aware devices that have been configured to perform KMI transactions transparent to users and operators. An authorized KMI end entity inserts a KMI token into the workstation and accesses the PRSN Common Ordering Manager. KMI and entities may choose to access a catalog of all online KMI product and services offerings in the KMI library. They also are offered a menu of templates for each KMI service and product for which they have been assigned a privilege. The templates are tailored to limit selection to only those options to which they have been granted privileges. They can either retrieve an existing request through the template and modify the data for resubmission or access a blank template. Once the request is completed, they submit it to the PRSN.

Tracking orders is performed in a similar manner. Each order is given a tracking number that can be referenced. An authorized operator can access a list of all pending orders. They can choose to query for status, update, or cancel a request. They also can choose to remain online while the status is requested, or select to have the PRSN send a notification of the action when it is available.

Distribution

This process arranges for the transfer of KMI products from the KMI to end users or intermediaries in a secure and authenticated manner. Two basic types of KMI products are distributed. The first type includes physical products (e.g., hard copy codebooks, canisters of hard copy key materials, and tokens). These are distributed through protected shipping channels (e.g., the Defense Courier System). A goal of the Target KMI is to reduce the amount of these materials to the extent operationally acceptable. The preferred means of distribution is protected electronic delivery. When a KMI product is available for distribution, it can be "pushed" automatically to the intended recipient. The PRSN includes an electronic vault for intermediate storage of black KMI products that have been generated previously. The KMI provides a capability for authorized users to "pull" materials from the vault. The vault also serves as a rapid access source for products that the KMI will deliver (or "push") to end entities.

Key Recovery

Key recovery capabilities allow a means for authorized KMI users to access KMI products associated with an encryption process (e.g., KRI) in the event that key is lost or otherwise unavailable. Two general applications exist for key recovery. One application is to enable local information owners to access information that is protected when a key is lost. The other is a central capability to provide KRI to other authorized individuals based on national policies for key recovery.

Revocation

Revocation is used in normal operations as individual responsibilities and privileges change, resulting in the need to invalidate individuals' KMI roles and privileges. It is also a critical component of recovery in the event that sensitive KMI materials of an individual, a KMI manager, or an internal KMI operation have been or are suspected of being compromised. The process for requesting a revocation is performed in the same manner as KMI product and service ordering. An authorized KMI manager inserts a KMI token into the workstation and logs into the PRSN. The KMI manager's workstation will access the Compromise Recovery Agent within the PRSN, which will validate the manager's identity and the role and privileges associated with that identity. The KMI manager is also offered a menu of intuitive templates to allow a revocation request to be accomplished. The templates are tailored to limit selection to only those options that they have been granted privileges. The KMI processes that request, and activates mechanisms automatically to prevent any operations using the revoked KMI materials.

8.1.7.3.4 System Operations— An Internal KMI Perspective

Although the previous section highlighted critical KMI system operations from the perspective of KMI users and managers, this section provides an overview of the internal operations of the Target KMI to support system functions. The KMI is designed to provide a set of common management functions to provide a uniform, consistent, and intuitive interface to KMI users and managers.

KMI manager and end-user workstations are structured as "light clients," using commercial Web technologies to support transactions with servers provided in the PRSN. This allows system enhancements to focus on updates to these servers, minimizing reconfiguration of RM software. From the perspective of the KMI internal operations, the KMI end entity uses a workstation and KMI token to access the PRSN. The connection is secured using the token as a basis for establishing identity and securing the transactions. The PRSN Access Manager validates the end entity's identity, role(s), and privileges before access is granted to any other KMI resources. For all operations, each server within the KMI will verify the privileges for the identity represented in the token and whenever feasible will provide tailored screens with pull-down menus for the entity to select any authorized operation desired. Archiving of audit information for all interactions will be maintained automatically by the PRSN. Tools will be available to allow authorized users and managers to query the audit information.

Registration

Registration by its nature requires involvement of users and operators. Registration allows an individual or device to receive a PKI identity. The RM accesses the PRSN and logs into the Registration Server. Using screen menus tailored for registration of the type of entity being registered, the RM enters the required identity information. The workstation, via the PRSN, accesses the external repository for information to be validated. It presents this information to the RM, annotating possible discrepancies. Once the RM accepts the identity as valid, the workstation develops an identity certificate.

Several concepts are still being considered for processes at this point. The scheme currently used is for the token to generate a public and private key pair. Other options are for the end user workstation or the CA to generate the pair. When the token generates the pair, the token transfers the public component to the RM workstation that, in turn, forwards it along with a certificate request through a PRSN for registration to a PKI CA PSN.[4] The PRSN assigns a KMI unique identifier to the identity. The CA creates and signs an identity certificate, updates the appropriate directory, and returns the certificate to the RM workstation. The RM workstation loads the certificate onto a token and the RM issues the token to the user. All tracking and audit information is performed automatically by the PRSN and CA PSN, as appropriate.

Enrollment

Using a KMI token to establish identity, the PM accesses the PRSN and logs into the PRSN Enrollment Server. Enrollment allows an individual or device to receive encryption keys. The PM then inserts the token for the end entity being assigned KMI roles and privileges. The Enrollment Server provides menu screens for the PM to select the operations desired. This includes the update of role definitions, privilege assignments to roles, and identities assigned to roles. All PM interactions will be automated and updated into the KMI library repository that stores enrollment status information.

Request and Tracking

An authorized KMI user or manager can access the PRSN and log into the Common Ordering Manager to request KMI products and services and to obtain status of requests that are being processed. The Common Ordering Manager will provide tailored, intuitive screens and will be validated against known data domains of the template and privileges of the product requestor.

[4] In selected operations, the private key is transferred in a secure manner to the CA (via the PRSN) to support future key recovery operations. Private keys associated with identity certificates are NOT escrowed.

Feedback to users is provided online if those checks find discrepancies before a request is accepted. The same basic sequence is used to cancel or update orders. When a valid request has been submitted, the Common Ordering Manager assigns an internal order tracking number and prepares an electronic order request.

KMI-aware devices incorporate capabilities to automatically and directly interact with the Target KMI. In this regard, they can initiate KMI requests automatically, interacting with the PRSN Device Ordering Manager function in a manner similar to the process used by authorized KMI users and managers. They will have to be registered as a valid end-entities and enrolled to authorize appropriate KMI privileges. Because there is no operator in the loop, they will not go through screens; rather they will generate requests in an automated manner. Orders from devices will be tracked in a standard manner so that device sponsor or component administrator can query status and intercede to update or cancel orders generated by devices under their purview.

KMI Product Generation

All KMI products will be generated within a PSN in response to order requests from a PRSN. These can result from product requests from KMI managers, directly from KMI-aware devices, or from event services. PSNs produce all physical KMI products. For electronic products, PSNs will provide only Black materials. The PSN will perform all cryptographic functions necessary to generate KMI products, to protect them while being processed and stored within the PRSN, and for distribution directly to an end entity or through an intermediary (such as a Communications Security [COMSEC] Custodian).

Delivery

PSNs arrange for delivery of all physical KMI products through proper physical distribution systems. However, the preferred distribution for KMI products is via Black electronic transfers. The Target KMI is structured to enable delivery directly to end entities, including KMI-aware devices that can interact automatically with the KMI. This presumes that the KMI-aware devices include appropriate protocols to facilitate the transfers and internal cryptographic processing.

As discussed earlier, the PRSN Delivery Agent server can push products, automatically initiating an electronic transfer of Black KMI products over a secure link to a designated recipient. Authorized recipients can access the PRSN and log into the Delivery Agent server to “pull” KMI products. The Delivery Agent establishes a secure link with the intended recipient and electronically transfers the Black KMI products over that link. PRSNs include a capability for an electronic vault, providing a repository for previously generated and encrypted products, each with a unique identifier, split into a nonsensitive portion that is stored, and a sensitive portion that is encrypted. The PRSN is capable of querying to determine the status of materials that are stored, deleting stored materials, and retrieving them. If KMI products have to be decrypted (e.g., to make additional copies that can be prepared for delivery to multiple end entities), to facilitate delivery, the products are transferred back to a PSN for additional processing, and the requisite Black products are returned to the vault.

Key Recovery

The KMI Key Recovery Agent capability will collect and archive all KMI information that may be needed to support key recovery operations. KRI will be encapsulated in a manner to require multiple approved KRAs to collaborate to gain access the sensitive KRI. The encapsulation will enforce protection and access controls resultant KRI as dictated by appropriate national policies (e.g., two or more pre-selected individuals will need to be involved to gain access to the unprotected KRI materials.) When KRI is accessed, it will be protected to prevent inadvertent disclosure and transferred onto a KMI token for delivery.

Revocation

Revocation of KMI privileges is accommodated by modification or deletion of roles and privileges as addressed under enrollment. The KMI will be able to revoke any KMI product. Each product will have a unique identifier (e.g., Certificate Number, Key Identifier). Authorized KMI managers can access the PRSN and log onto the Compromise Recovery Agent capability to process requests for revoking KMI products. When a validated request has been processed, the Compromise Recovery Agent will task an appropriate PSN to add the identified KMI materials to an appropriate mechanism to enforce the revocation.

The Target KMI will support two approaches for enforcing revocation. For certificate-based transactions, the KMI will integrate Online Certificate Status Protocol (OCSP) into their online validation servers. These servers provide worldwide distribution and access to information needed to ensure that only valid keys and certificates are being used. Protocols within KMI-enabled and KMI-aware applications and devices may include verification using these servers to show KMI materials at both ends of the transactions are valid. The Target KMI will also support the use of Compromise Recovery Lists, including CRLs and CKLs as other mechanisms. These support other than certificate-based operations and are for use in situations in which ready access to distributed, online servers is not operationally feasible (either based on mission constraints ala tactical environments) or at times when network access is limited or unavailable (e.g., network outages).

System Management

Each KMI site has provisions for a site manager to perform a number of critical operations. A primary responsibility of these managers is to manage the day-to-day operations of the site. The site manager is responsible for monitoring the performance of the overall site, and when necessary off-loading operations to another site as a backup capability. This includes a variety of tasks such as starting up, backing up, aborting, and restoring site operations. Other critical responsibilities are related to managing the security of the site, including operating intrusion detection systems (IDS), providing local site responses to intrusions, managing local security audits, sanitizing the site, and returning the site to a secure state. Site managers are responsible for coordinating the installation, testing, maintenance, configuration, and control of all components within the site.

The CSN is also responsible for managing the overall KMI. In addition to its own site management, it provides long-term archive capabilities, performs audits, provides help desk capabilities, and enforces and verifies the compliance of operations with established security policies. The CSN is also responsible for managing all KMI IDS reporting, analyzing the aggregated information, and formulating and coordinating responses to suspected and actual cyber attacks.

Each PRSN site has several subsystems that provide databases and data management services for the enclave. For example, each site will maintain the appropriate product catalog, registration data, and role and privilege data for clients that request products and services at that site. Each PRSN will also serve as a hot standby (backup) capability for other PRSNs and will have the capability for automated transfer of services to and from other PRSNs. Each PRSN also maintains a library of documents that can be downloaded by client node components and software modules that may be run by clients that access the PRSN.

The PSNs also have system management responsibilities unique to their sites. As production nodes, they have to plan and schedule production activities (based on historical demand statistics and customer demand projections), monitor production flows, and allocate production resources to best satisfy production demands. To support tracing and status reporting of orders, the PSNs perform accounting and tracking of all orders from time of order receipt—through each production stage—until transfer of Black materials back to the PRSN or delivery of physical products directly to recipients. Because the PSNs process sensitive KMI product materials, they will have to maintain archive capabilities to augment those in the centralized long-term CSN archive, tools to facilitate appropriate audits, and facilities and procedures to comply with KMI security policies.

8.1.7.3.5 Transition

Although the actual KMI structure will evolve over time, the KMI program has established a fundamental philosophy for transition. Enhanced system capabilities will be introduced in parallel with existing operational capabilities. The strategy will be based on NO HARD CUTOVER whenever feasible. This will allow users to plan and implement effective transition of their operations to take advantage of new capabilities. Legacy capabilities will be dismantled only after a complete operational transition has been accomplished.

Impact of Transition on KMI Clients

Transition from the present systems to the Target KMI and the interim transitions from one KMI CI to another are planned and will be executed to minimize the impact on KMI managers and users. The Target KMI architecture itself has been designed to be consistent with this tenet. One example is the use of a “light client” concept to allow KMI manager workstations to remain stable, with enhancements being introduced in the servers typically provided in PRSNs. Another example is the use of validation servers to perform security-critical certificate path validation and enforcement of compromise recovery as a means for providing a more stable environment for client applications.

The PKI capabilities plan to follow this to the fullest extent feasible. The adoption of commercial industry standards and trends will maximize the use of commercially available applications. Reliance on commercial PKI tool kits for enabling of DoD custom applications will ease PKI-enabling. However, commitment to commercial industry standards implies that custom DoD applications may have to be upgraded to follow the commercial sector's evolution. If custom applications incorporate special features to support DoD-unique requirements, the diversity of COTS and GOTS systems can create significant issues.

Broader KMI capabilities will also continue to evolve. However, the KMI will maintain its full complement of products and services, and introduce new capabilities as additions rather than replacements. As discussed above, KMI products and services will be dismantled only when the community no longer requires them. KMI tool kits will evolve to ensure backward compatibility and interoperability with the newest features of the KMI. KMI device owners, developers, and providers will have the opportunity to retain current operational configurations or take advantage of KMI advanced features, as they become available. The KMI's longer range capability increment rollout planning enables device developers to plan their products' evolution in an organized and efficient manner.

8.1.7.4 U.S. Federal Public Key Infrastructure

The Federal PKI is headed by the Federal PKI Steering Committee (SC), which is composed of representatives from all federal agencies either using or considering the use of interoperable public key technology in support of electronic transactions. The Federal PKI SC is chartered under the Enterprise Interoperability and Emerging Information Technology Committee of the U.S. Federal Government CIO Council. It also has strong ties to the Security, Privacy, and Critical Infrastructure Committee. It provides guidance to federal agencies and executive agents regarding the establishment of a Federal PKI and the associated services.

The Federal PKI SC also receives recommendations from the Federal PKI Technical Working Group (TWG), which responds to issues presented to it by the Federal PKI SC relating to the technical implications of developing the PKI.

The Federal PKI will support secure Federal Government use of information resources and the National Information Infrastructure (NII). The Federal PKI will establish the facilities, specifications, and policies needed by federal departments and agencies to use public key based certificates for information system security, electronic commerce, and secure communications.

The Federal PKI will support secure communications and commerce among federal agencies; branches of the Federal Government, state, and local governments; business and the public. The Federal PKI will facilitate secure communications and information processing for unclassified applications.

The Federal PKI will be created largely from the bottom up. Federal efforts to use public key cryptography begin with individual applications within agencies that provide immediate support for vital agency programs. These implementations are paid for largely out of program funds, not funded as a centralized Government PKI.

The core Federal PKI consists of CAs, RAs, certificate status responders, and management authorities that manage public key certificates used by federal departments and agencies for unclassified application.

PKI clients will use the public key certificates issued and managed by the PKI to provide security services to federal users, such as key pair generation, digital signature generation, digital signature verification, and confidentiality key management.

The Federal PKI is fielding a BCA that provides certification paths between CAs in agencies and outside the Government. Federal CAs that meet the requirements of the Federal Bridge Certificate Policy will be eligible to cross-certify with the BCA, thereby gaining the certification paths needed for broad trust interoperation in the larger federal and national PKI. Certificates issued to and from the Federal BCA will normally include certificate policy mapping extensions that allow relying parties to establish that remote certificate policies are equivalent to local ones. The Federal BCA operates under the control of the Federal PKI Steering Group, which is the Certificate Policy Authority for the Federal Government. Establishing policy mapping equivalencies is one of the Federal Policy Authority functions.

One driver of the Federal BCA design was the need to accommodate hierarchical and mesh PKI implementations that are already common within the Federal Government. Both hierarchical and mesh PKIs are operated by U.S. Federal Government commercial and government partners. The BCA concept enables applications capable of processing mesh PKI certificates to interoperate with any mesh or hierarchical PKI cross-certified with the BCA.

Some commercial clients already include the certificate path development and validation capabilities needed to take advantage of the BCA. Other vendors are now upgrading their PKI client applications with the features necessary to operate with the BCA. Figure 8.1-20 illustrates the planned architecture of the Federal PKI.[5]

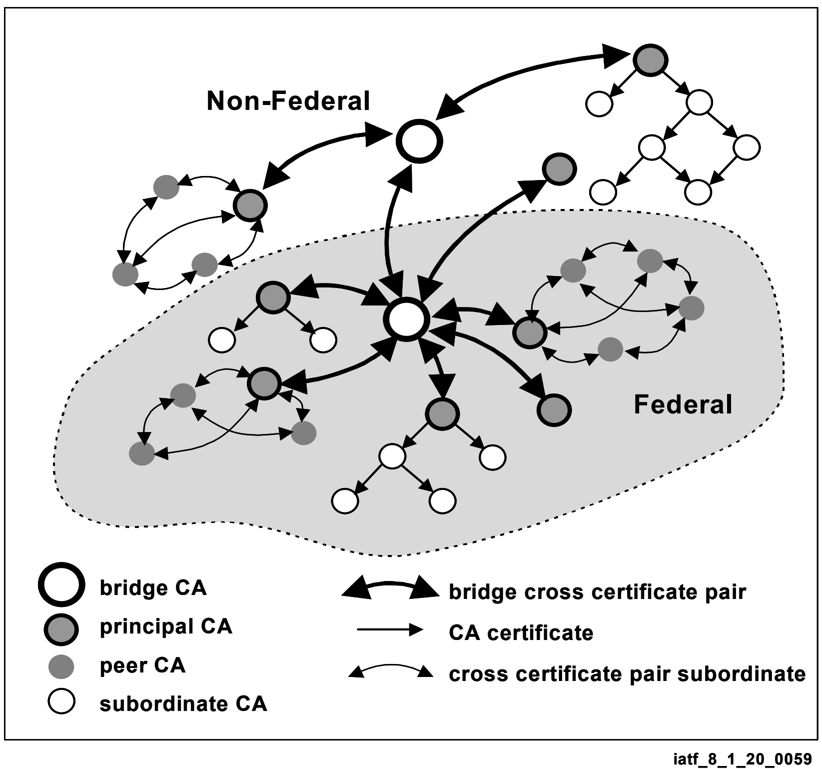

Figure 8.1-20. Federal PKI Architecture

5 Figure 8.1-20 courtesy of the Federal PKI Web page at http://csrc.nist.gov/pki/twg/welcome.html.

The BCA will actually consist of a variety of CA products that are mutually cross-certified. This design allows several vendors to operate within the BCA "membrane," thus allowing for continued BCA operation in the face of a dynamically changing PKI technology and vendor environment.

8.1.7.5 Corporate PKI

8.1.7.5.1 Introduction

This section describes how the Microsoft Information Technology Group (ITG) built a PKI by deploying a hierarchy of CAs hosted on Microsoft Windows 2000 servers. The name of this project was the Crypto Management Architecture PKI. In this discussion, it will be shortened to CMA PKI.

8.1.7.5.2 Requirements

Microsoft is implementing and using many security technologies to protect and maintain the integrity of digital intellectual property. A large number of these security technologies depend on the use of valid X.509 certificates issued by trusted CAs.

The CMA PKI must support the deployment of the technologies listed in Table 8.1-5 to satisfy the corresponding business requirements:

Table 8.1-5. Business Requirement and Security Technology Comparison

Business Requirement	Security Technology
Employees in all Microsoft business units need to exchange encrypted and/or digitally signed e-mail with each other, external business partners, and customers over the Internet and other untrusted networks	Secure Multipurpose Internet Mail Extensions (S/MIME)
Secure networking with a common transport/tunnel technology supported by uniform authentication architecture	Internet Protocol Security (IPSec) and Layer 2 Tunneling Protocol (L2TP)
Users must be able to store encrypted data securely, whereas the corporation must be able to recover data should an employee leave or lose his/her encrypting certificate	Encrypting file system (EFS) and EFS recovery policies
Reduce the costs of purchasing certificates from outside sources by providing internally generated certificates for all intranet and most extranet SSL servers.	Secure Sockets Layer (SSL) or Transport Layer Security (TLS)
Strong authentication	Smart cards
Replace the practice of giving various external business partners shared corporate network accounts by trusting certificates from vendors and business partners	Certificates
Nonrepudiation	Digital signatures

Additional requirements are as follows:

- Active Directory integration (e.g., CRLs, certificate enrollment, certificate templates, and CA certificates available via Active Directory).
- Certificates mapped to users and computers in Active Directory.
- Servers and client computers automatically enrolled for certificates (i.e., autoenrollment).
- Interoperability with Exchange Key Manager Server (KMS) and Outlook.
- A healthy foundation for the expansion of Microsoft's corporate PKI to support forthcoming confidentiality, integrity, and authentication features in Microsoft products.

8.1.7.5.3 PKI Design

The Inherited PKI

At the beginning of the CMA PKI project, Microsoft already had a PKI managed by Legal and Corporate Affairs (LCA) and Product Release Services (PRS). This PKI, which was developed to support various product group and manufacturing efforts, was not used for general corporate functions.

Because Microsoft's root authority (MSROOT) in the inherited PKI is the top of the company's certification hierarchy for digitally signing all of its software products, a compromised MSROOT would have very negative national and global consequences. Therefore, the CAs that make up the inherited PKI are located in a secure vault on the Microsoft campus. The vault cannot be entered by a single individual; rather, it must always be entered by two authorized individuals simultaneously. The vault also has been designed to withstand attacks by cutting torches, explosives, and other brute force tools of nefarious individuals.

CMA PKI Topology

The CMA PKI has CAs in a three-level rooted hierarchy:

- **Level 1: Microsoft Corporate Root Authority.** The root CA at the top level of the hierarchy signs its own certificate. ITG makes it available to all entities that may want to establish trust in it.
- **Level 2: Microsoft Intranet CA and Microsoft Extranet CA.** The CAs below the root CA in a three-level hierarchy are referred to as policy CAs or intermediate CAs. These CAs have certificates issued from the root CA and can be online or offline; ITG chose to keep the intermediate CAs offline for security reasons.
- **Level 3: Microsoft Intranet CAs.** The third level in a rooted hierarchy contains the issuing CAs. An issuing CA, as the name implies, issues certificates to end-entities.

Issuing CAs are normally online CAs—in other words, they are always connected to the network.

Certification Authority Servers

To establish the CMA PKI, eight CAs needed to be built. Three of the new CAs are off line and reside in the LCA vault. The other five CAs will be online and service requests 24 hours a day, 7 days a week. These servers will reside in the ITG vault.

Microsoft Corporate Root Authority

The Microsoft Corporate Root Authority is a Windows 2000 CA. This represents the top of the Corporate PKI and is used only to sign/certify subordinate CAs. This server will be off line, except with generating revocation lists or signing CAs, and will reside in the current LCA vault. This server should be built with the following parameters:

- Windows 2000 Certificate Server (Stand alone Root CA).
- Self-signed CA certificate.
- Hardware-based Crypto Service Provider (CSP).
- 8-year CA lifetime.
- 2,048 CA key length.
- 90-day CRL publishing interval.
- CRL locations: LDAP to Active Directory; HTTP to crl.microsoft.com.

Microsoft Intranet CA

The Microsoft Intranet Certification Authority will certify all other CAs used for internal purposes. This server will be off line except with generating revocation lists or signing CAs, and will reside in the current LCA vault in. This server should be built with the following parameters:

- Windows 2000 CA (Stand alone Subordinate CA).
- Install certificate from a PKCS#7 text file from the Microsoft Corporate Root Authority.
- Hardware-based CSP.
- 5-year CA lifetime.
- 2,048 CA key length.
- 90-day CRL publishing interval.
- CRL locations: LDAP to Active Directory; HTTP to crl.microsoft.com.

Microsoft Intranet Network CA

The Microsoft Intranet Network Certification Authority will issue end-entity certificates for services that relate to general server, user, or network administration, such as Administrator certificates, EFS recovery certificates, router (IPSec/L2TP) certificates, and smart card enrollment agent certificates. The servers comprising this CA will be continuously on line, will require redundancy, and will reside in the ITG vault. These servers should be built with the following parameters:

- Windows 2000 CA (Enterprise Subordinate CA).
- Install certificate from a PKCS#7 text file from the Microsoft Intranet CA.
- Hardware-based CSP.
- 2-year CA lifetime.
- 2,048 CA key length.
- 24-hour CRL publishing interval.
- CRL locations: LDAP to Active Directory; HTTP to itgweb.corp.microsoft.com.

Microsoft Intranet FTE User CA

The Microsoft Intranet User Certification Authority will issue end-entity certificates to full-time employees (FTE) users on the corporate network for general client authentication, EFS, and smart card logon. The servers comprising this CA will be continuously on line, require redundancy, and reside in the ITG vault. These servers should be built with the following parameters:

- Windows 2000 CA (Enterprise Subordinate CA).
- Install certificate from a PKCS#7 text file from the Microsoft Intranet CA.
- Hardware-based CSP.
- 2-year CA lifetime.
- 2,048 CA key length.
- 24-hour CRL publishing interval.
- CRL locations: LDAP to Active Directory; HTTP to itgweb.corp.microsoft.com.

Microsoft Intranet Non-FTE User CA

The Microsoft Intranet User Certification Authority will issue end-entity certificates to non-FTE users on the corporate network for general client authentication, EFS, and smart card logon. The servers comprising this CA will be continuously on-line, require redundancy, and reside in the ITG vault. These servers should be built with the following parameters:

- Windows 2000 CA (Enterprise Subordinate CA).
- Install certificate from a PKCS#7 text file from the Microsoft Intranet CA.
- Hardware-based CSP.
- 2-year CA lifetime.
- 2,048 CA key length.
- 24-hour CRL publishing interval.
- CRL locations: LDAP to Active Directory; HTTP to itgweb.corp.microsoft.com.

Microsoft Extranet CA

The Microsoft Extranet Certification Authority will certify all other CAs used for external purposes. This server will be off line except with generating CRLs or signing CAs and will reside in the current LCA vault. This server should be built with the following parameters:

- Windows 2000 CA (Stand alone Subordinate CA).
- Install certificate from a PKCS#7 text file from the Microsoft Corporate Root Authority.
- Hardware-based CSP.
- 5-year CA lifetime.
- 2,048 CA key length.
- 90-day CRL publishing interval.
- CRL locations: LDAP to Active Directory; HTTP to crl.microsoft.com.

Microsoft Personnel E-Mail CA

The Microsoft Personnel E-Mail Certification Authority will issue end-entity certificates used for digitally signing and encrypting email (S/MIME). The server hosting this CA will be continuously on line, require redundancy, and reside in the ITG vault. These servers should be built with the following parameters:

- Windows 2000 CA (Enterprise Subordinate CA).
- Install certificate from a PKCS#7 text file from the Microsoft Extranet CA.
- Hardware-based CSP.
- 2-year CA lifetime.
- 2,048 CA key length.
- 24-hour CRL publishing interval.
- CRL locations: LDAP to Active Directory; HTTP to crl.microsoft.com.

8.1.7.6 Other Implementations

Kerberos Solution

Kerberos provides another approach for IA and network security. [7] Kerberos is a network authentication protocol designed to provide strong authentication for client/server applications by using secret-key cryptography. A free implementation of this protocol is available from the Massachusetts Institute of Technology (MIT). Kerberos also is available in many commercial products.

The Internet is an insecure place. Many of the protocols used in the Internet do not provide any security. Tools to “sniff” passwords off the network are in common use by systems crackers. Thus, applications sending an unencrypted password over the network are extremely vulnerable. Worse yet, other client/server applications rely on the client program to be “honest” about the identity of its users. Other applications rely on the client to restrict its own activities with no additional enforcement by the server.

Some sites attempt to use firewalls to solve their network security problems. Unfortunately, firewalls assume incorrectly that the hackers are on the outside. Insiders carry out most of the really damaging incidents of computer crime. Firewalls also have a significant disadvantage in that they restrict how users can use the Internet. Firewalls are simply a less extreme example of the dictum that there is nothing more secure than a computer that is not connected to the network—and powered off! In many places, these restrictions are simply unrealistic and unacceptable.

Kerberos was created at MIT as a solution to these network security problems. The Kerberos protocol uses strong cryptography so that a client can prove its identity to a server (and vice versa) across an insecure network connection. After a client and server have used Kerberos to prove their identities, they can also encrypt all of their communications to assure privacy and data integrity as they conduct their business.

Kerberos is freely available from MIT, under a copyright permission notice very similar to the one used for the Berkeley Software Distribution (BSD) operating system and X11 Windowing system. MIT provides Kerberos in source form, so that anyone who wishes to use it may look over the code to assure themselves that the code is trustworthy. In addition, for those who prefer to rely on a professional supported product, Kerberos is available as a product from many different vendors.

In summary, Kerberos is another approach to network security problems. It provides the tools of authentication and strong cryptography over the network to help secure your information systems across the entire enterprise.

References About Kerberos

- More information about Kerberos can be found on the Internet at http://www.isi.edu/gost/info/kerberos
- An excellent introductory article can be found at http://www.isi.edu/gost/publications/kerberos-neuman-tso.html

8.1.7.7 Additional References—Supporting Documentation on the Target KMI

As discussed in the roadmap, the Target KMI will be realized in an evolutionary manner through a series of CIs. The definition of the Target KMI provides a perspective for each CI to ensure that the goals established for it will be achieved. Specifically, the Target identifies the physical nodes, allocates the functionality within each node, and specifies the physical interface standards for KMI external and internal boundaries. The KMI program has an ongoing systems engineering activity to define and plan the Target KMI definition. In January 2000, the KMI Program published a series of documents that describes the Target KMI definition that resulted from those activities. These documents include the following:

- KMI 2010, Overview and Summary Information.
- KMI 2001, Mission Needs Statement (MNS).
- KMI 2002, KMI Operational Requirements Document (ORD).
- KMI 2000, Functional Requirements Document.
- KMI 2022, Standards and Technology Assessment.
- KMI 2020, System Interface Description.
- KMI 2003, KMI Security Policy and Requirements.
- KMI 2004, KMI Threat Assessment Report.
- KMI 2005, KMI System Security Architecture.
- KMI 2006, KMI Security Risk Analysis/Assessment.
- KMI 2012, Operational View (CONOPS).
- KMI 2011, Program Glossary.
- KMI 2021, Use Case Package (Five Volume Document).
- KMI 8000, Target Architecture Validation Report.

8.1.8 Future Trends of Public Key Infrastructure

PKI is one of the most promising technologies on the horizon today to provide strong authentication, data integrity, confidentiality, and nonrepudiation services to a wide user base. The evolution of PKI has been dynamic, and this trend will assuredly continue into the future. Although PKI products have been on the market for years, the technology still lacks maturity. Much work remains to be done by product vendors and implementers. In addition, the public awareness of the benefits of PKI needs to be heightened before PKI will become the "silver bullet" it is intended to be.

One ongoing problem with PKI is incompatibility among vendor solutions. PKI standards need to continue to be developed and proven. Although several major PKI vendors are in the marketplace, many of the current products do not work with those from another vendor. However, many vendors use the specifications provided by the RSA Security Company, which have become in some cases, de facto standards.

There has been a growing trend toward standardization for certificates and cryptographic token storage formats. Through technical exchange meetings with vendors and standards groups, such as the ETF and ITU, it is likely that officially recognized standards will eventually be approved. These standards are vital for PKI to meet the demands for IA. PKI vendors have recognized this, and competing companies have shown increasing willingness to work together to produce common standards.

As better standards emerge, PKI products will improve. For example, the RSA PKCS #12 certificate container format allows private keys and certificates to be stored in a file on a disk. Access to this information can be protected by a password. Because the user chooses the password, a bad password choice can impair the security of the stored certificate. Despite its disadvantages, the PKCS #12 format is the most widely used format and, at present, there is no widely available alternative that is suitable for replacing PKCS #12. An improved method is needed, but a new method needs to be accepted by the entire industry to become successful.

Vendors that produce interoperable products allow enterprises to purchase PKI equipment with less fear that the purchased product will become obsolete or will no longer be supported. Instead, it is known that the product will operate with others, even if the products are produced by a competitor. When the time comes to upgrade, upgrades are less painful if mature standards are in place. The upgrade can be phased in over time, and there should be a richer set of upgrade features from which to choose. A wide variety of unrelated applications could benefit from a common security solution. A common system reduces the long-term costs associated with maintaining a separate security infrastructure for each application. PKI could provide this solution, and interoperability among a common PKI is important.

Over time, the underlying cryptography of PKI will need to change continuously. It is obvious that new computers are constantly becoming faster. Faster computers will benefit the "brute force" method of cracking encrypted information. As such, the encryption technology must improve to stay ahead. As consumer computers are able to process data to crack the current encryption scheme in a reasonable period of time, data protected by cryptographic techniques becomes more endangered. Even without advances in computer speed, advances in other areas, such as distributed computing, will make encryption upgrades a requirement. A PKI integrator should not assume that a major investment in PKI would be a one-time expense.

8.1.8.1 Smart Cards

One of the promising new PKI implementations will be smart cards, which will provide vast new advantages for PKI. Private keys will be stored in a microchip on the card rather than on a computer disk. The smart card contains not only data, but also a microprocessor to manipulate and protect the stored data. The smart card can control access to private key on the card, and prevent unauthorized manipulation of the data.

Once the private key has been generated by a smart card, the onboard crypto-processor contains the private key. This processor prevents outside access to the private key. Smart cards also offer an alternative to the limitations of the RSA PKCS #12 certificate container by providing additional security to the private key.

Smart cards will provide mobility to PKI users. A single card could be used for physical access to a building, to log in to a computer, and to securely transmit information.

There are some disadvantages to smart cards. An obvious disadvantage is that they might easily become lost or stolen. Although a stolen smart card should not reveal any information to its finder, its legitimate owner might not have a means to access his computer system or might gain access to a building. Another disadvantage of smart cards is that it may be desirable to operate several computer systems, each of which employ a smart card. If a user has only one smart card or does not have enough to use simultaneously, the smart cards will not be useful.

8.1.8.2 Biometrics

Biometric devices represent another emerging technology. These devices use physical features or behavior characteristics of human beings to identify a person. Biometric devices will measure unique qualities, such as a person's retina or fingerprint. Upon login, the devices measure the appropriate qualities of the user and then compare those qualities with known qualities, which are stored digitally.

The technology is advancing rapidly. When combined with PKI and smart cards, biometrics offer additional advantages. PKI alone cannot guarantee the identity of a person. The person using the PKI usually enters a password or PIN to access the private key and to identify himself or herself to the PKI. If this password is compromised, the PKI is compromised. Instead of using a password, a user could use a biometric device to authenticate himself or herself to a PKI system via a biometric device. The biometric device provides additional assurance that the person is actually who he or she claims to be. The addition of biometrics is a solution when assurance of authentication to the PKI is essential.

At present, well-chosen and protected passwords can provide a higher level of assurance than biometric devices because of their lower probability of being guessed versus the higher probability of a biometric device mistakenly identifying a person. As biometric devices improve, their accuracy is likely to improve significantly. Biometric devices offer increased value by taking some risks out of user passwords. Examples of password risk include users choosing simple, easily guessed passwords, or users writing passwords on a piece of paper that is not properly secured.

Biometrics is expected to grow significantly in the security field within the next 10 years. Although prices are still relatively high, biometric devices will come down in time. Several companies are already marketing biometric devices to the public. The combination of biometrics with PKI provides synergy between these two technologies. Biometrics provides a more secure login than a simple password access to one's private key, and PKI allows biometric devices to be used across a wide system infrastructure. Disadvantages to biometrics include not only the users' resistance to the biometrics requirement that their personal qualities (e.g., retina image) be examined or stored, but also the relatively high cost of the biometric devices.

8.1.8.3 Certificate Revocation

A certificate revocation scheme needs to be in place to prevent a user's certificates from being valid when a PKI user has his or her access to the PKI removed. For example, an employee who leaves a position or is transferred to another position will likely need to have access removed. Because this user's public key may still exist in the local directories of other users' computers, a method needs to be in place to prevent the certificate from being used. Two leading methods are being investigated to accomplish this effort: CRLs and the OCSP.

CRLs are a comprehensive record of all certificates that have been issued previously but are no longer valid. The CA publishes, and is responsible for, the CRL. The CRL includes the serial numbers of all certificates that have been revoked. This scheme requires a client wishing to check a certificate against a CRL to download the entire CRL. The CRL would then be searched to discover if any listed certificates match the certificate that the client is checking. An expiration date is included in the CRL, at which time the CRL must no longer be replied on for validation.

The OCSP is another method to ensure the currency of a certificate. A work in progress by the IETF, it employs a client/server approach. A client wishing to validate a certificate sends a request to a server. The request includes a list of certificates or serial numbers that the client wishes to check. The server sends back a reply, which is signed by a CA to ensure the validity of the reply. The reply has several possible responses: Not Revoked, Revoked, On Hold, or Expired.

At present, there is no clear consensus on which method will prevail. Certificate revocation schemes will be a major task of future PKI development.

8.1.8.4 Certificate Recovery

A key recovery system might be employed on some PKIs. The recovery system allows access to the private key through an alternate means. For example, this is useful if the user forgets a password, or management must know the contents of a user's encrypted message. Key recovery systems may be appropriate for encryption keys, but are not recommended for identity keys.

Identity keys are used only for identity purposes. For example, when a user wishes to add nonrepudiation benefits to an e-mail message, the user can sign the e-mail with a private identity key. An encryption key is used to provide confidentiality services. If the user wishes to send a confidential e-mail message, the public encryption key of the addressee would be used. The private key of the addressee is required to view the message contents.

A key recovery system will allow the encrypted data to be made available to the trustees of the key recovery system. Because an identity key is only used to provide identity services, there is no legitimate reason to recover the key. If the password to the identity key is lost, the key can be revoked and a new key issued. A PKI policy can help prevent the misuse of identity keys to falsely impersonate a user by not permitting identity keys to be escrowed in a key recovery system.

Key recovery systems have serious security ramifications. Introducing a key recovery system into a PKI introduces a weak link in the security chain. Although key recovery systems can be a method to help guard against dishonest users, there is no guarantee that a person entrusted with the key recovery system will not be dishonest as well. A security breach in this system could remove virtually all of the security advantages of PKI. In the future, biometric devices might help prevent the lost password problem. If key recovery systems are still desired for other reasons, they should be employed with great care.

8.1.8.5 KMI

The KMI is a common structure to administer keying material within DoD. The KMI will eventually administer all keying material throughout DoD. This material includes legacy symmetric key products and public (asymmetrical) key products. As KMI becomes a common administration tool for all DoD keys, it will be used for key registration, key generation, secure key archiving, and key distribution. Additional systems are being examined to study the feasibility of integrated into the KMI.

The KMI architecture will likely consist of several nodes. A CSN will provide data storage, a root certificate authority, archive audit records, and IDSs. A PSN will provide key generation services and certificate generation at the CA level. A PRSN will provide key registration, tracking, directory services, key recovery services, and privilege assignment. The clients node will distribute keys and provide an interface for customer services.

The CSN is envisioned to have KMI databases and library services. It would provide support to supervise the KMI system and could operate at the Secret level. The PSN will likely be designed with a modular construction. The key generation and management functions can be added or deleted as they are needed. The PSN would support new services as they become available. The PRSN would be deployed on the DoD networks (e.g. NIPRNet, SIPRNet) and would be intended for deployment regionally. The PRSN, which would operate at the network's classification level, would provide support for key recovery services within the KMI.

The KMI will likely need to be accredited to operate at system high. Various nodes will operate at, for example, Top Secret-high and Secret-high, as needed. Provisions will be need to in place to isolate nodes with dissimilar classifications and to prevent data cascading to a lower classification. In the future, it is possible that the DoD KMI will interface with other KMIs within the United States and with its allies. Policies will need to be changed to allow crypto data transmission over protected LANs such as SIPRNet.

A KMI and a PKI are closely related technologies, that are designed to work together. The KMI will provide support for the keys that the PKI must use. The PKI program benefits by making use of an existing key infrastructure, while providing new capabilities. According to the NSA KMI Standards and Technology Survey, key management will be accomplished in a similar method to that developed for multicast groups. Policies are constructed for numerous groups. Group keys are created by a group controller, which then distributes them. The Group Secure Association Key Management Protocol (GSAKMP) is then used to distribute the groups' policies and provide for future rekeying of each group when needed.

The KMI is a work in progress. The plans for the system will likely change as it is designed and built. At present, it is uncertain how the KMI will be modified or what additional users it will eventually serve.

8.1.8.6 Risks Associated with this Analysis

This analysis of what PKI will be like in the future consists of predictions based on current trends today. The PKI momentum has been building for several years and is likely to continue. However, PKI has shown fairly slow growth so far. The growth is not widespread at present outside a few select industries. As standards and new technologies mature, PKI will likely become much more important.

There are several risks in predicting the future trends of PKI. Usability will be an extremely important factor in the PKI maturation. Although important advances in this area have been made, more will need to occur in the future. It is also possible that another technology will emerge that can provide similar benefits and will be more efficient to deploy. At present, the future of asymmetric keys to provide strong authentication, data integrity, confidentiality, and nonrepudiation services appears to be solid. PKI is the technology most likely to benefit from the advantages of asymmetric keys to provide these services.

8.1.8.7 Conclusions

PKI can be expected to grow vigorously in the next 5 to 10 years. As standards are developed and more applications are supplied with PKI built in, the PKI will grow more quickly. It is possible that one or more competing technologies also will arise on the security scene, but such a technology will likely provide similar capabilities that PKI promises. The advantages of PKI will be the flexibility to adapt to new applications and to provide a common security architecture that can be deployed for many applications, involving both computers and other devices.

The future of PKI will depend largely on its usability. Even the best security resources cannot provide security if they are not accepted by end users. PKI offers numerous benefits and is intended to be used for more than one application. For example, an e-mail system may use PKI for confidentiality and nonrepudiation across an enterprise and to operate with external enterprises. A database system might use the same PKI to provide confidentiality and nonrepudiation plus authentication to the database. As more applications use a common PKI, additional economies of scale can be realized. Existing applications will need to be replaced with newer software that is PKI compliant, or PKI enabled. Application integration will likely be one of the most difficult and most expensive phases of adopting a common PKI system.

References

1. International Telecommunication Union (ITU), 1997, Information Technology—Open Systems Interconnection—The Directory: Authentication Framework, ITU-T Recommendation X.509.

2. RSA Laboratories, November 1, 1993, PKCS#10: Certification Request Syntax Standard, Version 1.0.

3. RSA Laboratories, November 1, 1993, PKCS#7: Cryptographic Message Syntax Standard, Version 1.5.

4. Bruce Schneier. Applied Cryptography, pp. 139-152.

5. PKIX –4. Public-Key Infrastructure (X.509) (pkix), August 7, 2000 http://www.ietf.org/html.charters/pkix-charter.html.

6. PKI Profile. NIST PKI Program February 23, 2000, http://csrc.nist.gov/pki.

7. Massachusetts Institute of Technology's Kerberos: The Network Authentication Protocol Web Site, June 24, 2000, http://web.mit.edu/kerberos/www.

Additional References

a. Furlong, Judith, Public Key Infrastructure (PKI) Scenarios Overview, November 20, 1997.

b. University of Southern California The Kerberos Network Authentication Service http://www.isi.edu/gost/info/kerberos.

c. B. Clifford Neuman and Theodore Ts'o. Kerberos: An Authentication Service for Computer Networks USC/ISI Technical Report number ISI/RS-94-399. September 1994, http://www.isi.edu/gost/publications/kerberos-neuman-tso.html.

d. The Moron's Guide to Kerberos, Version 1.2.2, http://www.isi.edu/gost/brian/security/kerberos.html.

8.2 Detect and Respond as a Supporting Element

A fundamental tenet of the defense-in-depth strategy embraced by this Information Assurance Technical Framework (IATF) is to prevent cyber attacks from penetrating networks, and to detect and respond effectively to mitigate the effects of attacks that do. An integral aspect of this strategy is a secure infrastructure to support the detection of and reaction to cyber incidents and attacks.

8.2.1 What This Focus Area Addresses

Detect and respond capabilities are complex structures that run the gamut of intrusion and attack detection, characterization, and response. The progression of detect and respond technologies is building from audit logs and virus scanners to a more robust capability. While technology continues to evolve, this overall area remains heavily dependent on highly skilled operators and analysts.

8.2.1.1 Scope of This Focus Area

The local environments (within an enclave) are the logical location for network-based and host-based sensors. Sections 6.4, Network Monitoring within Enclave Boundaries and External Connections, 6.5, Network Scanners within Enclave Boundaries, and 7.2, Host-Based Detect and Respond Capabilities within Computing Environment, address specific Framework guidance for these sensors. This section addresses the processes and technologies that are typically required beyond the sensors. This includes discussions of architectural considerations for improving the Detect and Respond posture of an enterprise, evolving paradigms for a Detect and Respond infrastructure, the various processes and functions that are performed within the secure infrastructure, and the technologies that are available to realize these processes and functions. The section concludes with sources for additional information and a list of references used in developing this guidance.

8.2.1.2 Terminology

To set the stage for the discussions in this section of the Framework, there are a number of terms that should first be defined. We recognize that these terms, which are fundamental to the discussions in this section, are also germane to many sections of the Framework. We also appreciate that these terms have varying interpretations within the community, so we include the following definitions to eliminate possible confusion or ambiguity within this section of the Framework.

The first set of terms deals with threats and vulnerabilities. A threat exists when an intruder (also referred to as an adversary or a threat agent*)* has the means, motivation, and opportunity to exploit an information system and/or its associated networks. A vulnerability is a weakness or hole that can be exploited by an intruder. An attack is a sequence of events an intruder uses to exploit a vulnerability.

An intrusion can be thought of as a break-in attempt or actual break-in to an information system. The intruder's intent may be to misuse the system or data contained within the system, render a system unreliable or unusable, gain access to the data contained on the system, and/or manipulate the data. Once an intrusion has occurred on an information system, the damage can be extensive—sensitive information may be compromised and network systems or network services can be rendered inoperable. These events can result in the loss of a corporation's competitive edge, lost productivity when network services are unavailable, and costly man-hours and dollars to assess the impact of an intrusion and recover any lost data.

Beyond this, there are various levels of an "attack" that are also worth identifying. We look at attacks from a bottom up perspective, since they are detected based on a logical progression from the point of view of sensors (e.g., intrusion detection system or IDS).

- Alarms are the typical output provided by a sensor as an indication that it believes it detected some evidence of the presence of an intruder.
- Events are actual occurrences of some irregularity that caused an alarm. We distinguish alarms from events in that there are often a number of valid network and host operations that may cause an alarm (thus giving rise to false positive indications).
- Interesting Events are based on the recognition that local environments may experience hundreds of thousands of events daily, and there are typically only a small number that have the potential for any real damage. This category represents those that have the potential for serious impact such as may be characterized in a security policy.
- Incidents are interesting events that actually have serious impact on the information systems and networks of a local environment.
- Attacks are concentrated efforts by an adversary or intruder to have a serious impact on an overall enterprise, usually implemented by a series of incidents targeted at multiple local environments.

While all incidents and attacks are important, the Framework guidance focuses on attacks in which the attacker(s) have the will, resources, and persistence to cause grave harm to an enterprise.

8.2.2 Enterprise Architecture Considerations

While planning for a Detect and Respond infrastructure, it is important to recognize that the enterprise networks and systems that it will support must also be structured to provide information to, and take advantage of, the services and information such a secure infrastructure provides. The remainder of this section provides guidance on configuring an enterprise to improve its Detect and Respond posture.

Incident Reporting

As highlighted in Sections 6.4, Network Monitoring within Enclave Boundaries and External Connections, 6.5, Network Scanners within Enclave Boundaries, and 7.2, Host-Based Detect and Respond Capabilities within Computing Environment of the Framework, the local environments have the option of deploying sensors, and possibly analysts, to interpret the results of, and, when appropriate, react to the implications of these outputs. Beyond the local environment, each organization, or perhaps community, has to determine what information should be reported, in what format, under what situations, and to whom. The Department of Defense (DoD) has issued implementation guidance and a joint policy for incident and vulnerability reporting. Other system infrastructures simply allow reporting, and leave it to the local environment to work directly with the next tier to decide when, what, and how to report.

Network Partitioning and Redundancy, Backup

Networks are typically configured to provide the most cost-effective service to its users. Whenever feasible, networks should be partitioned into logical segments, with boundary protection devices between segments. This limits traffic flow and thus potential exposure within segments, provides a degree of isolation if one segment or another is subverted, and facilitates the shutting down or limiting of services within affected segments as a possible response. Offering redundant capabilities within a network creates the potential for response options allowing authorized traffic to be diverted around a segment that has been exploited.

Deploy Technical Safeguards and Countermeasures as Response Options

A fundamental aspect of an effective react capability is to deploy safeguards and countermeasures that can be activated to implement responses. Whether they are making changes to firewall policies, filtering router configurations, deception servers, or others, there are a number of such countermeasures available, as discussed in Section 8.2.5.4, Response Tools.

Plan for Contingency Operations

There is an entire discipline associated with disaster planning (sometimes referred to as planning for contingency operations) that includes the development of anticipatory processes and procedures that can facilitate an effective response. These include creating backups of mission-critical and establishing preplanned courses of action (COA). Recommendations regarding the preparation of COAs include the following:

- Plan to deal with high-probability threats and at least acknowledge the less likely possibilities.

- Allocate resources to complete and coordinate the planning; create plans in advance rather than waiting for an event to occur.
- Coordinate and obtain approval/acceptance of plans by upper management, business unit managers, and other decision-makers.
- Take advantage of planning that other, similar organizations may have already prepared.
- After the plans are formulated, exercise the procedures to validate the approach, refine the tactics, and train the participants.

When the program is in place, frequently review, update, and enhance it to keep it current.

Coordinating Responses

Fundamentally, response itself is an issue for the local environments. However, there are a number of factors with implications beyond the perspective of local sites that need to be considered when formulating and evaluating response options as well as when actually responding to an intrusion or attack. A basic decision is whether to shut down an intruder's access (or an entire site) or to allow an intrusion to continue while evidence is collected that will be needed for subsequent prosecution.

Considerations for Operations

As with the architectural features identified above, there are also complementary operational practices[1] that are important to the overall defense of an enterprise, and again, are directly relevant to considerations for a detect and respond infrastructure:

- Be prepared for severe denial-of-service attacks (e.g., institute and practice contingency plans for alternate services).
- Inspect for physical penetrations.
- Educate users and staff.
- Institute well-known procedures for problem reporting and handling.
- Institute procedures for reporting suspicious behavior.
- Institute and monitor critical access controls (e.g., restrict changeable passwords, require dial-back modems).
- Minimize use of the Internet for mission or time-critical connectivity.

---

[1] Note that it is imperative to perform quality network management and system security administration to maximize the security of the network configuration and mechanisms and to increase the likelihood of detecting and successfully reacting to attacks.

- Require security-critical transactions (e.g., establishing identity when registering) to be conducted in-person.
- Institute and monitor a strict computer emergency response team alert and bulletin awareness and patch program.
- Establish procedures for recovery from attack.

8.2.3 General Considerations for a Detect and Respond Solution

It appears that there are no generally accepted architectural constructs for a detect and respond infrastructure across various communities. However, there are several fundamental considerations for a detect and respond infrastructure that appear to be consistent across communities. These are highlighted below.

8.2.3.1 General Constructs for a Detect and Respond Infrastructure

In general, many network infrastructures are inherently hierarchical by nature, and this one is no exception. When considering a general construct for a detect and respond infrastructure, a primary consideration is the perspective that the system infrastructure layer will maintain for its support. Figure 8.2-1 identifies typical layers in this hierarchy and the perspectives that each layer could offer. Each layer usually retains responsibility for its own operation, and thus must be capable of making decisions about courses of action for its own operation. However, it is seldom the case that any site can function in a completely autonomous fashion without some oversight, coordination, and direction, so there is a natural hierarchy for the decision making as well.

In general, information about incidents, which is usually sensed at the lowest layer in the hierarchy, is reported to higher layers. Warning and response coordination that is more typically derived from higher layers is disseminated from these higher layers down. Again, these are general statements, and any specific situation has to be tailored to the unique needs of the constituent segments.

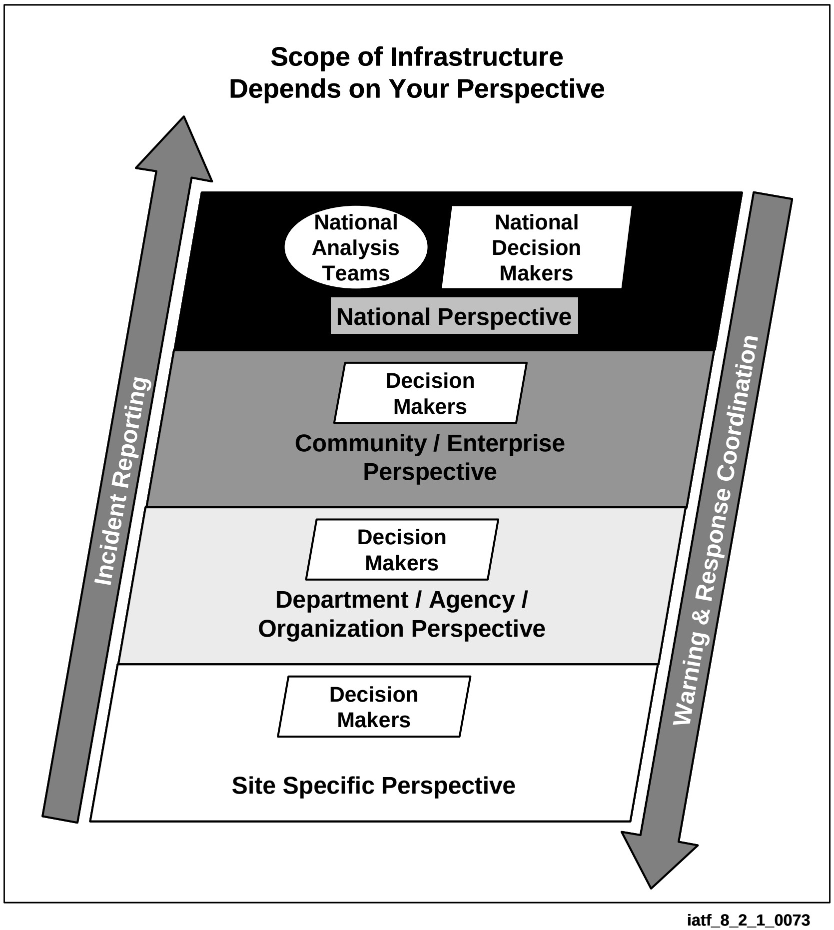

Figure 8.2-1. Perspectives of Layers in a Detect and Respond Infrastructure Hierarchy

8.2.3.2 Examples of Existing Detect and Respond Infrastructures

A detect and respond infrastructure of this nature will likely be structured in the manner depicted in Figure 8.2-2. This is consistent with various actual hierarchy structures used today in various communities and enterprises. The specific relationships and responsibilities across the layers differ in actual practice.

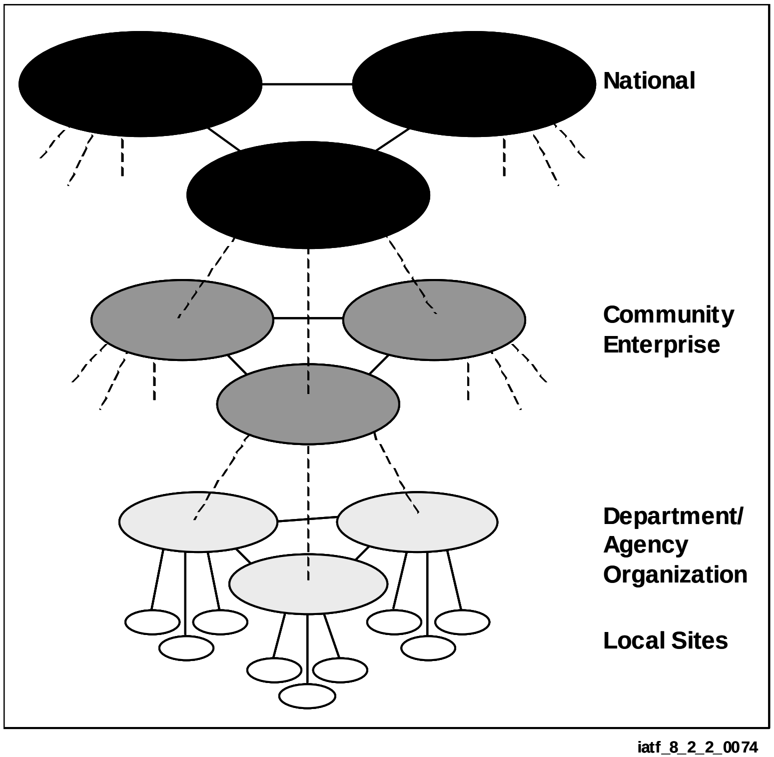

Figure 8.2-2. Basic Hierarchy for Detect and Respond Infrastructure

For the Department of Defense (DoD), local sites are responsible for deploying network monitors and performing site assessments. Typically each Military Department (MILDEP) has its own Navy Computer Emergency Response Team (NAVCERT) capability or Air Force Information Warfare Center (AFIWC) that is responsible for attack detection and characterization for that MILDEP. At the enterprise level, DoD has established a Joint Task Force for Computer Network Defense (JTF-CND), with a technical analysis capability within the Global Network Operations Security Center (GNOSC) to monitor critical defense networks and coordinate actions across the DoD to restore functionality after an intrusion or attack. The DoD model differs from the others in that reporting and response coordination procedures are mandated.[2]

The civil government agencies have adopted a less formal structure. There is a Federal Computer Emergency Response Team (FEDCERT) that is responsible for coordinating detect and respond activities across the Federal Government, but its use appears to be at the discretion of individual agencies. Selected agencies maintain their own Computer Emergency Response Team (CERT) capabilities (e.g., Department of Energy [DOE] Computer Incident Advisory Capability [CIAC] that is operated at Lawrence Livermore National Laboratories as a central clearinghouse for reporting incidents.) This community also takes some advantage of CERT capabilities from academia (e.g., CERT associated with Carnegie Mellon University actually

2 The DoD has issued CJCSI 6510.01B, a JCS publication providing implementation guidance and a joint policy for Defensive Information Operations. Within that document, Enclosure D, Appendix G, defines incident and vulnerability reporting procedures, methods, and reporting formats.

funded by DoD). The Federal Intrusion Detection Network (FIDNet), a General Services Administration (GSA) initiative to centralize a federal government-wide capability to analyze local sensor outputs is consistent with this general hierarchy but may be implemented as a managed commercial security service offering available to those agencies that decide to subscribe.

In the private sector, CERTs are available to support those specific organizations that choose to use them, again with reporting and coordination at the discretion of the organization. The Information Sharing and Analysis Center (ISAC), a construct resulting from efforts to implement Presidential Decision Directive 63 (PDD-63), was conceived as a mechanism to structure sector (e.g., banking and finance, telecommunications) coordinators. The intent was to provide a mechanism for enabling appropriate, anonymous, and confidential sharing of information on incidents, threats, vulnerabilities, and solutions associated with each sector's critical system infrastructures and technologies. One ISAC is in place for the banking and finance community. While others have not been put into operation, it is again representative of the use of a hierarchical structure for a detect and respond infrastructure.

At the national level, the National Infrastructure Protection Center (NIPC), established at the Federal Bureau of Investigation (FBI) again in response to PDD-63, is intended to serve as the U.S. Government focal point for threat assessment, warning, investigation, and response to threats or attacks against our nation's critical infrastructures. It is supported by the National Security Incident Response Center (NSIRC) at National Security Agency (NSA) to bring perspectives from the Intelligence Community to perform in-depth analysis (including post-attack investigation) to support activities at the NIPC (and JTF-CND). While these national layers of the infrastructure are called upon at the discretion of other organizations, they maintain a national-level perspective. The NIPC also leads or coordinates activities associated with national security or criminal investigations of cyber crimes.

Although not depicted in Figure 8.2-2, there is some evidence of global infrastructures being established at the international level. One such example is the Forum of Incident Response and Security Teams (FIRST), whose membership includes DoD Service CERTs, academia, and major private corporations from across the globe. Their goals are to foster cooperation among constituents for the protection, detection, and response from computer intrusions. They provide a means for sharing alert and advisory information, and facilitate collaborative planning and sharing of information, tools, and techniques.

8.2.4 Detect and Respond Functions

Within the detect and respond infrastructures, a wide range of functions is needed to support operations. In many cases, technology solutions are not available to perform these functions automatically. Analysts, network operators, and system administrators perform many of the functions by applying basic support technologies to ease their tasks. This section provides an overview of the functions that these analysts (with their tools) are attempting to perform. This section begins with an overview of the various phases of operation associated with detect and respond and then highlights specific functions that are representative of each phase. The section that follows provides a discussion of the underlying technologies that are available to support detect and respond capabilities.

8.2.4.1 Phases of Operation

Figure 8.2-3 illustrates the five basic phases of detect and respond. These phases are as follows:

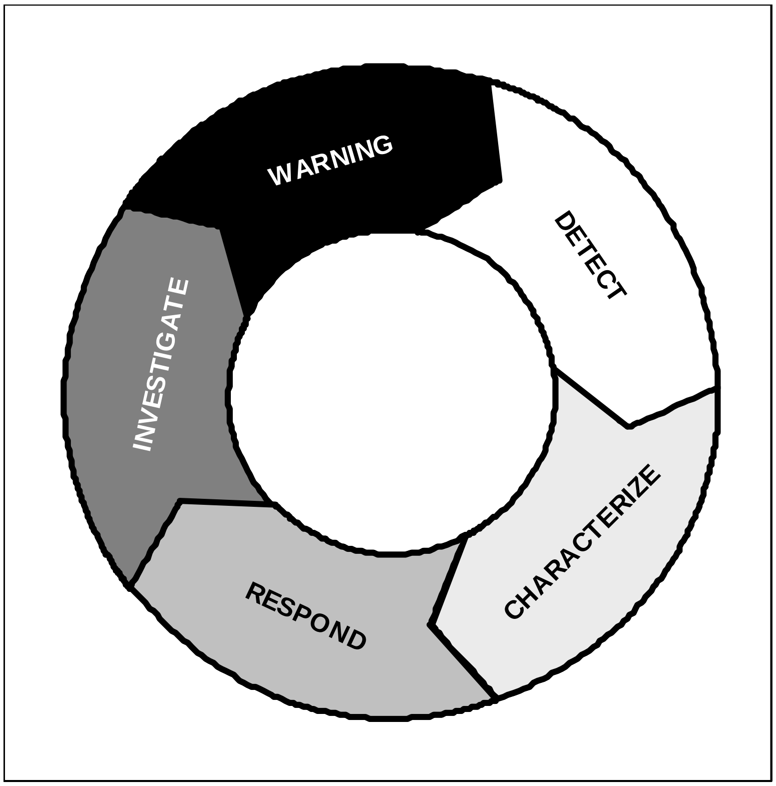

Figure 8.2-3. Basic View of Detect and Respond Phases

- **Warning—**Providing advanced notice of a possible impending attack, including a perspective on the attack strategy, scenarios, likely target sites, and timing
- **Detect—**Determining that an attack is occurring or has occurred. This includes the sensing functions discussed in Sections 6.4, Network Monitoring within Enclave Boundaries and External Connections, 6.5, Network Scanners within Enclave Boundaries, and 7.2, Host-Based Detect and Respond Capabilities within Computing Environment, of the Framework, along with broader activities to discern an attack is under way
- **Characterize—**Analyzing the attack in terms of its intent, approach, projections of how it will proceed, likely impacts, and possible identification of the attack source
- **Respond—**Reacting to mitigate the effects of the attack and restore the systems and network

- **Investigate—**Analyzing how an attack was accomplished to provide feedback to improve existing protect, detect, and react capabilities to ensure that similar exploitations cannot occur, and when appropriate, to provide evidence when prosecution of attackers is pursued.

From a process standpoint, it is possible to consider detect and respond operations as a series of phases or stages form a life cycle for a particular incident or attack. In this view, it is easy to consider the cycle of phases to begin anew with the occurrence of another attack.

While this perspective is straightforward, it is not really reflective of real-life situations. Although there is sense of “hand-off” from one phase to another, each of the phases is really an ongoing set of processes. For example, warning does not typically stop after an alert is issued.

It continues to search for new indications while detection capabilities focus on those being anticipated. This is typically the same for each phase, as represented in Figure 8.2-4. This sort of twisting view of detect and respond phases may seem whimsical, but is really more indicative of practical operations.

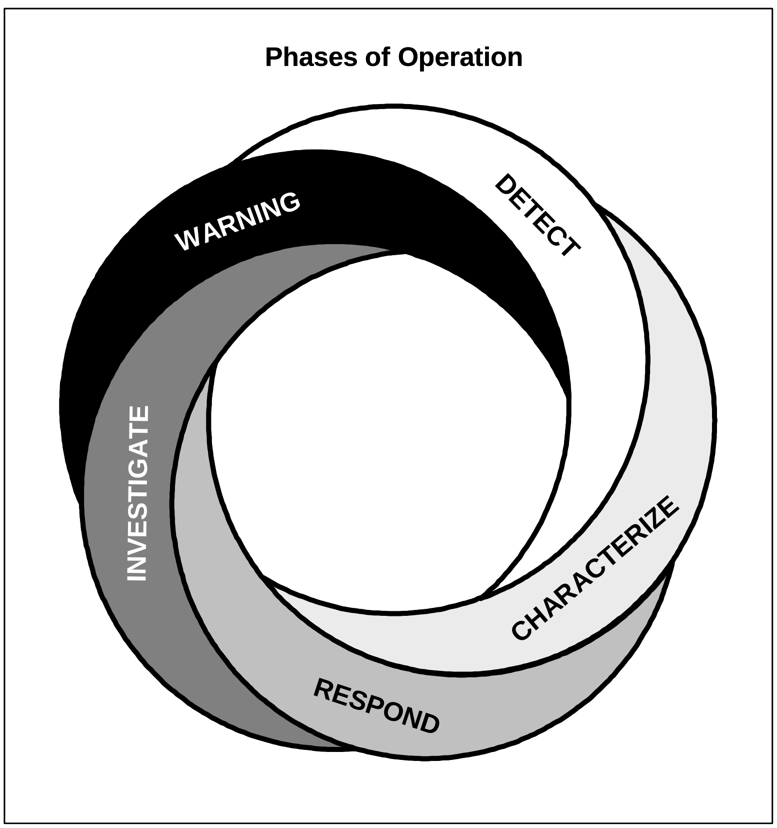

Figure 8.2-4. Realistic View of Detect and Respond Phases

There are a number of approaches for realizing these phases within the context of a detect and respond hierarchy. Figure 8.2-5 provides a perspective that can be used when considering allocation of detect and respond functions. While each local site, organization, or enterprise (community) has the option of allocating detect and respond functions within their hierarchy, it is often the case that warning and attack investigation is provided as a detect and respond infrastructure services because the investigation requires highly skilled analysts and access to broad and diverse sources of information. The other functions tend to follow the perspective on the hierarchy level. Thus, the functions on the left side of Figure 8.2-5 that focus on incidents are typical of those at a local level, or possibly an organizational level. Those on the right side of the diagram that focus on attacks are more indicative of those of a higher level of the system infrastructure (based on the view that attacks are really composed of coordinated incidents across multiple sites).

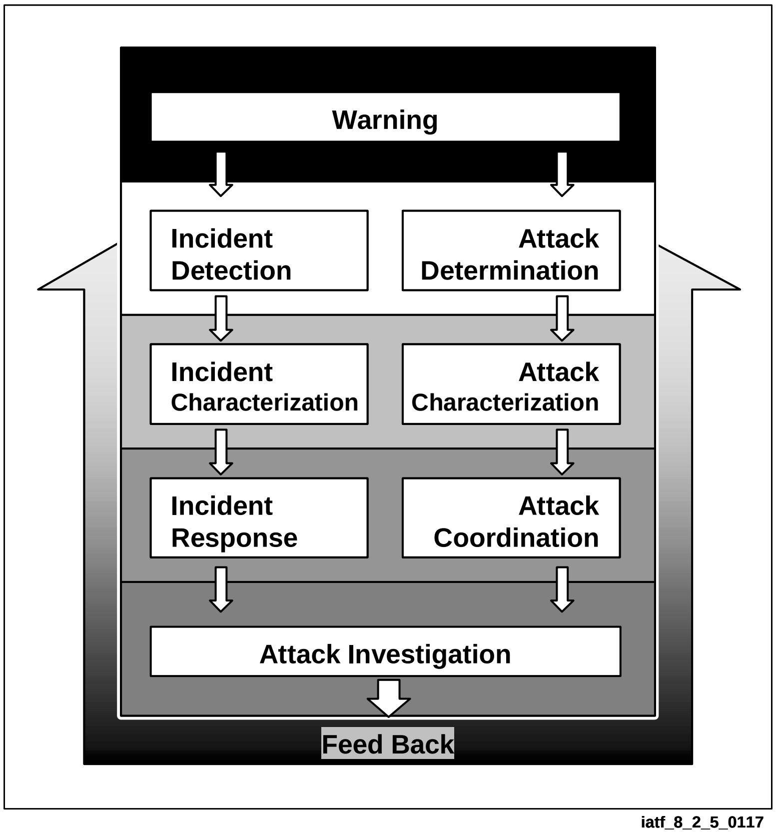

Figure 8.2-5. Possible Allocations of Detect and Respond Functions

Another important aspect of these functions is that they are highly dependent on one another. They each rely on, and provide information to others, working toward a common goal of successful detection and response to incidents and attacks. The following section highlights representative processes for each of the eight functions identified in the figure. Again, these are offered not as direction of what functions have to be performed, but to offer a perspective on what detection and response must achieve using the available technologies discussed in subsequent sections.

8.2.4.2 Functions to Support Warning

Warning is a proactive capability intended to provide advanced notice (or warning) of possible impending cyber attacks. Figure 8.2-6 offers a perspective on the types of functions that could be implemented to support warning.

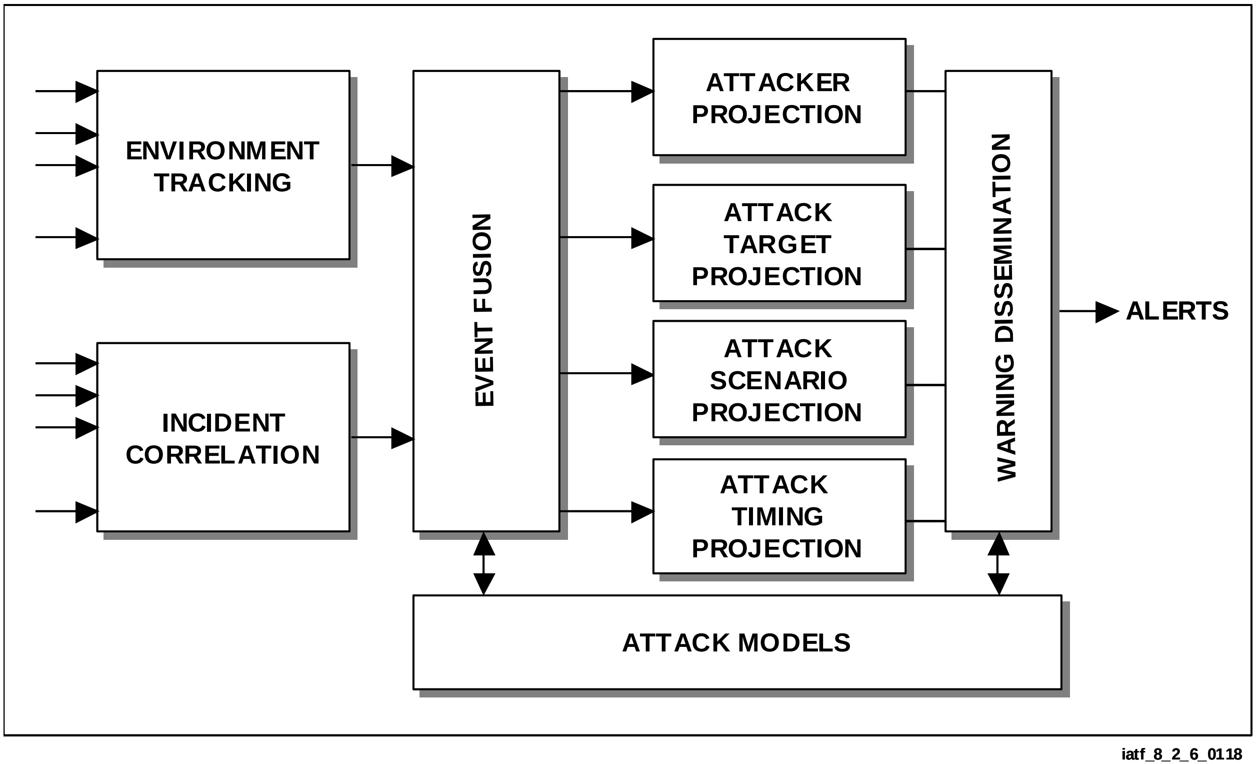

Figure 8.2-6. Functions to Support Warning

While this is undoubtedly a critical capability for maintaining an effective defensive posture, it is also the least mature. Discussion in the community seems to focus on the identification of precursors to attacks as "observables," tracking a broad range of social, political, organizational, intelligence, and technical events that can be fused with incident reporting to postulate attacker actions including attack target sites and systems and attack scenarios and timing. Various attack models are used as a foundation for these projections.

8.2.4.3 Functions to Support Incident Detection

Detection of incidents (or intrusions) is typical of a local site operation, as discussed in detail in Sections 6.4, Network Monitoring within Enclave Boundaries and External Connections, and 7.2, Host-Based Detect and Respond Capabilities within Computing Environment, of the Framework. In a broad sense, these functions at the local level are performed to determine the security posture and status of a local site (or environment) typically using network-based and host-based sensor technologies, supported by local analysts to identify vulnerabilities, intrusions, and malicious code attacks. Typical functions associated with support to local incident detection are shown in Figure 8.2-7.

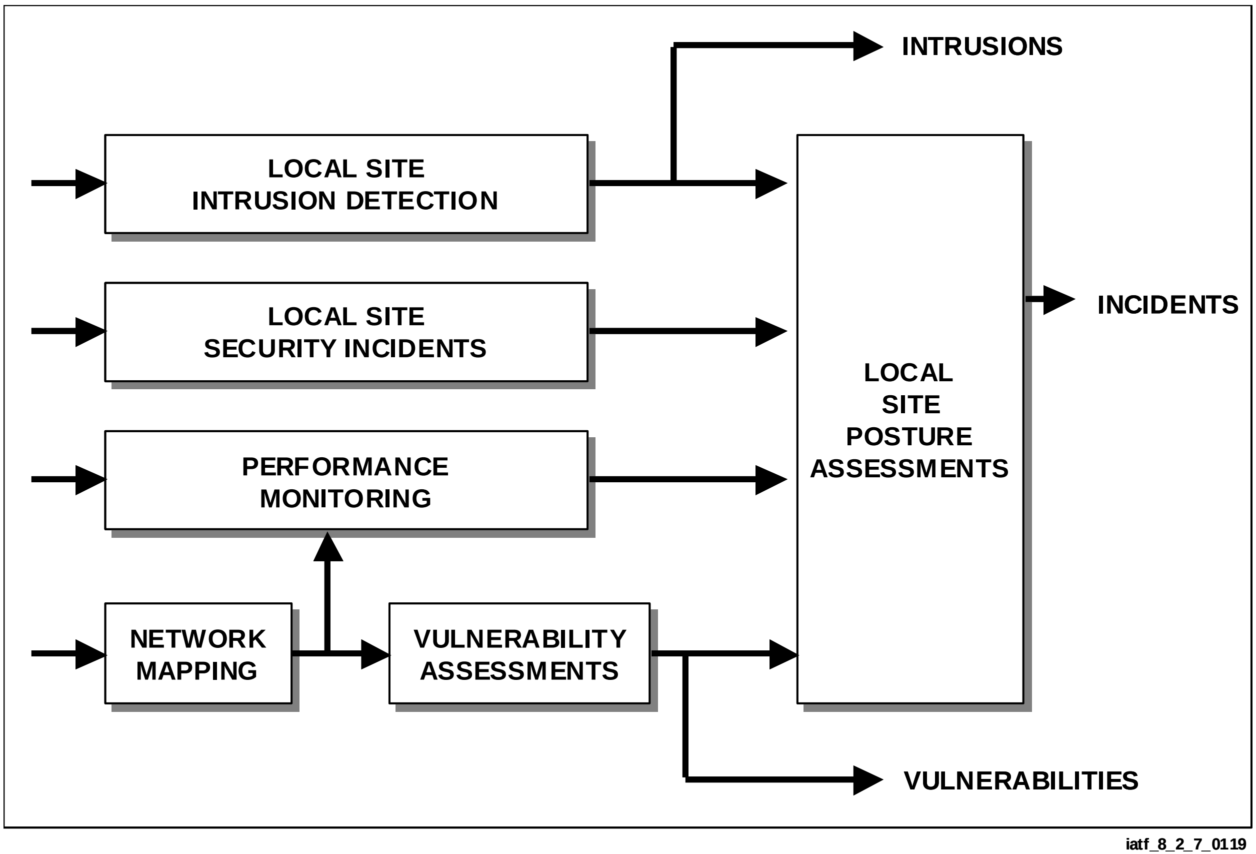

Figure 8.2-7. Functions to Support Local Incident Detection

To be consistent with other functional structures discussed in this section, we distinguish incident detection from incident characterization, in which operators perform analyses to discriminate between alarms, events, interesting events, and intrusions. As inferred by the diagram, these functions go well beyond intrusion detection to consider security incidents, performance irregularities, and vulnerabilities identified by scanners or penetration (e.g., Red Team) testing.

8.2.4.4 Functions to Support Incident Characterization

These functions draw from the results of the incident detection discussed in Section 8.2.4.3, Functions to Support Incident Detection, to interpret the true nature and criticality of each alarm that is created by the local sensors. Typical functions of incident characterization are shown in Figure 8.2-8.

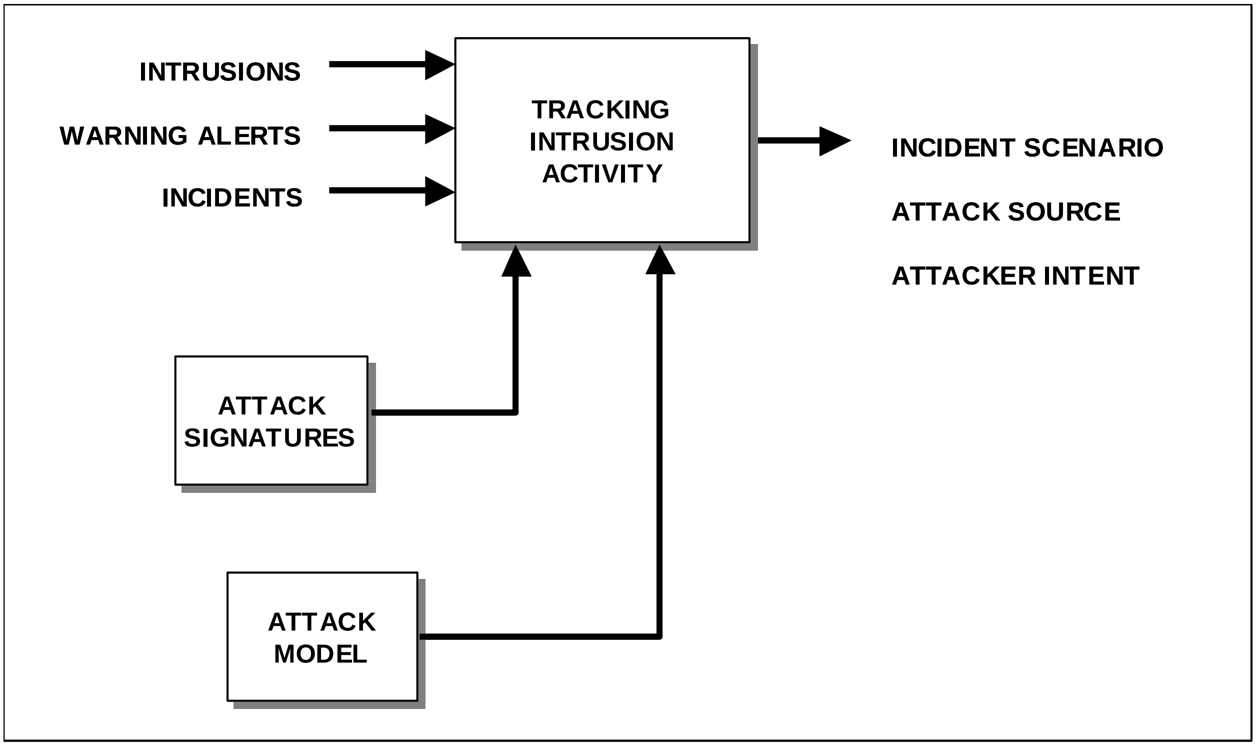

Figure 8.2-8. Functions to Support Incident Characterization

In addition to the primary inputs from incident detection, warning alerts provide an additional focus on specific attack sources and/or types of attacks. Ideally, the outputs of these functions would provide some sense of an intruder's intent, scenario, and the identification of the source of each incident. Typically, the results of these functions are used as input to the incident response functions, discussed below.

8.2.4.5 Functions to Support Incident Response

As discussed earlier, the local environment is ultimately responsible for executing a response to mitigate the effects of the intrusion and to restore the systems and networks. Typical functions of incident response are shown in Figure 8.2-9.

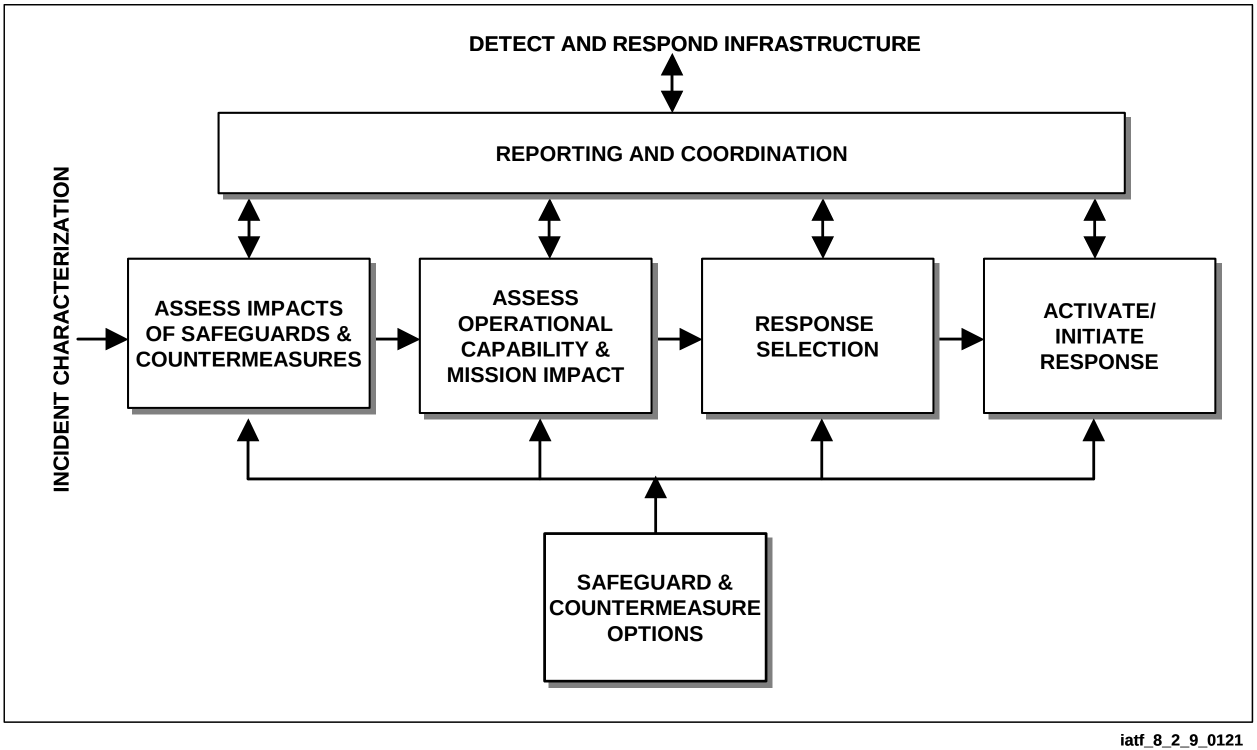

Figure 8.2-9. Functions to Support Incident Response

These functions draw from a set of preestablished safeguards and countermeasure options. Selection of an appropriate response option would be made based on a number of assessments. These assessments first address the impact (and any anticipated progressions) of the incident on the site's operational capabilities and its ability to perform its missions. The focus is then turned to how the activation of available responses would impact the site's operational capabilities and ability to perform its missions. Coordination with the detect and respond infrastructure (when appropriate) can provide recommendations about the technical impacts that response options may have on incidents associated with ongoing attacks as another factor for consideration in selecting a response. Finally, these functions include the activation of the selected response, intended to contain, assess damage, eradicate, reconstitute, and recover from the effects of the incident (or attack) to the local site capabilities.

8.2.4.6 Functions to Support Attack Determination

Building on intrusion and incident reporting from local sites and external events, these functions focus on determining if an attack is under way or has occurred. Typical functions associated with attack determination are shown in Figure 8.2-10.

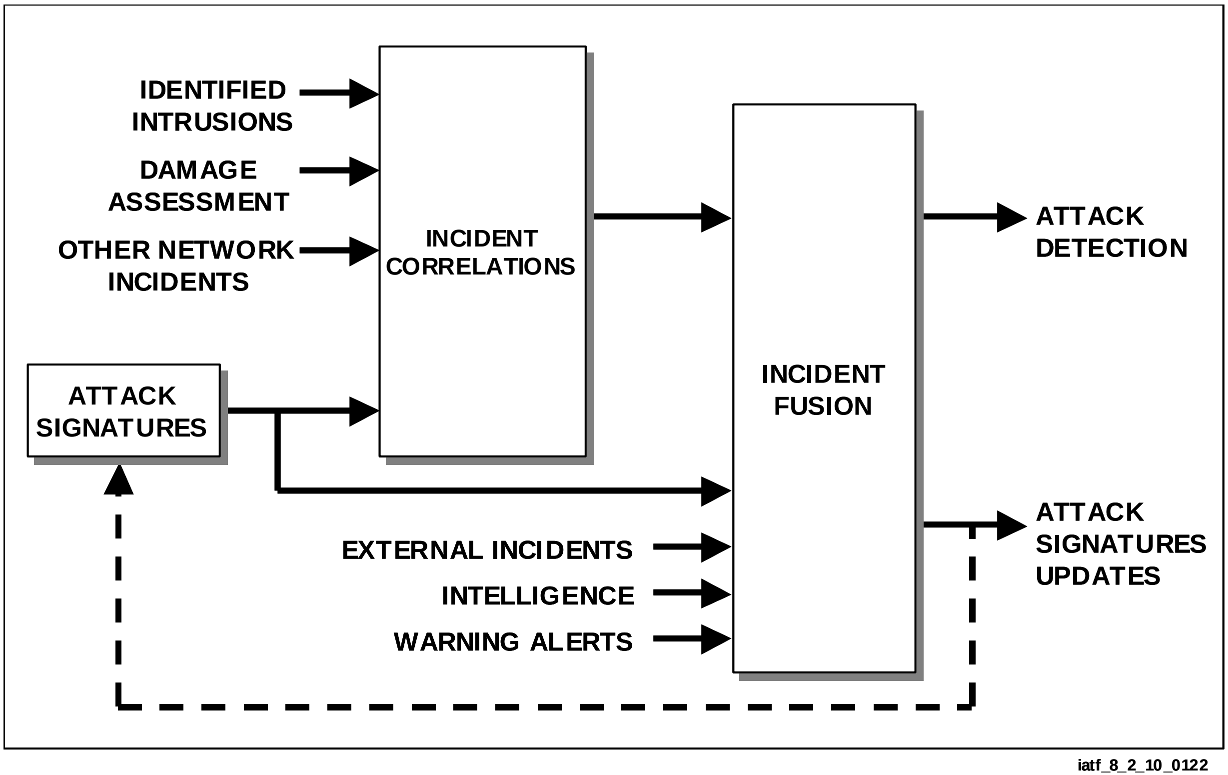

Figure 8.2-10. Functions to Support Attack Determination

Drawing from the local sensing functions discussed in Sections 6.4, Network Monitoring within Enclave Boundaries and External Connections, 6.5, Network Scanners within Enclave Boundaries, and 7.2, Host-Based Detect and Respond Capabilities within Computing Environment, of the Framework, this activity also includes correlation of incident data from all sites within its constituency and combining that data with warning alerts, all-source intelligence reports, and other external events to discern if an attack is under way.

8.2.4.7 Functions to Support Attack Characterization

When the determination has been made that an attack has been detected, this set of functions focuses on analyzing the attack in terms of its intent, approach, projections of how it will proceed, likely impacts, and possible identification of the attack source. Typical functions associated with attack characterization are shown in Figure 8.2-11. The functions can be considered in two categories. The first is fusion of the various sources of information to identify all relevant events and data to be analyzed. The second is a series of specific analysis functions that focus on the various aspects of the characterization.

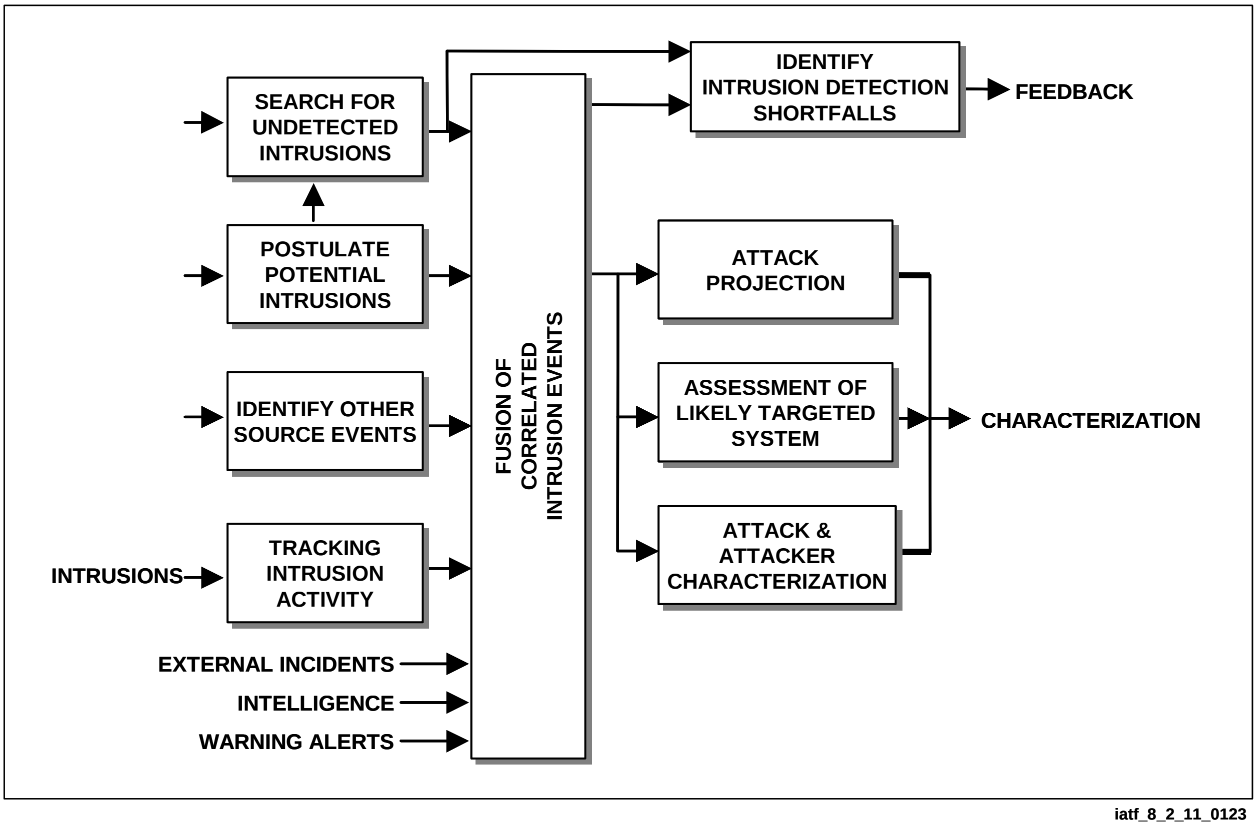

Figure 8.2-11. Functions to Support Attack Characterization

Resources available to support analysis include warning alerts, all-source intelligence, external incidents, known attack scenarios, and attacker signatures and electronic fingerprints. A side benefit of these analyses is feedback that can be provided to local IDSs to support their tuning, updating their attack scripts, and the like, to improve their detection capabilities as they pertain to the ongoing attack.

8.2.4.8 Functions to Support Response Coordination

When an attack has been detected and characterized, the real value the system infrastructure can provide is coordinating an effective response at the local sites that will mitigate the effects of the attack and support the restoration needed to return the systems and networks to normal operation. Typical functions associated with response coordination are shown in Figure 8.2-12. The thrust of these functions is to assess, on a technical (versus operational and mission impact) basis, the effectiveness of available preplanned courses of action, safeguards, and countermeasures against the identified and projected attack scenarios.

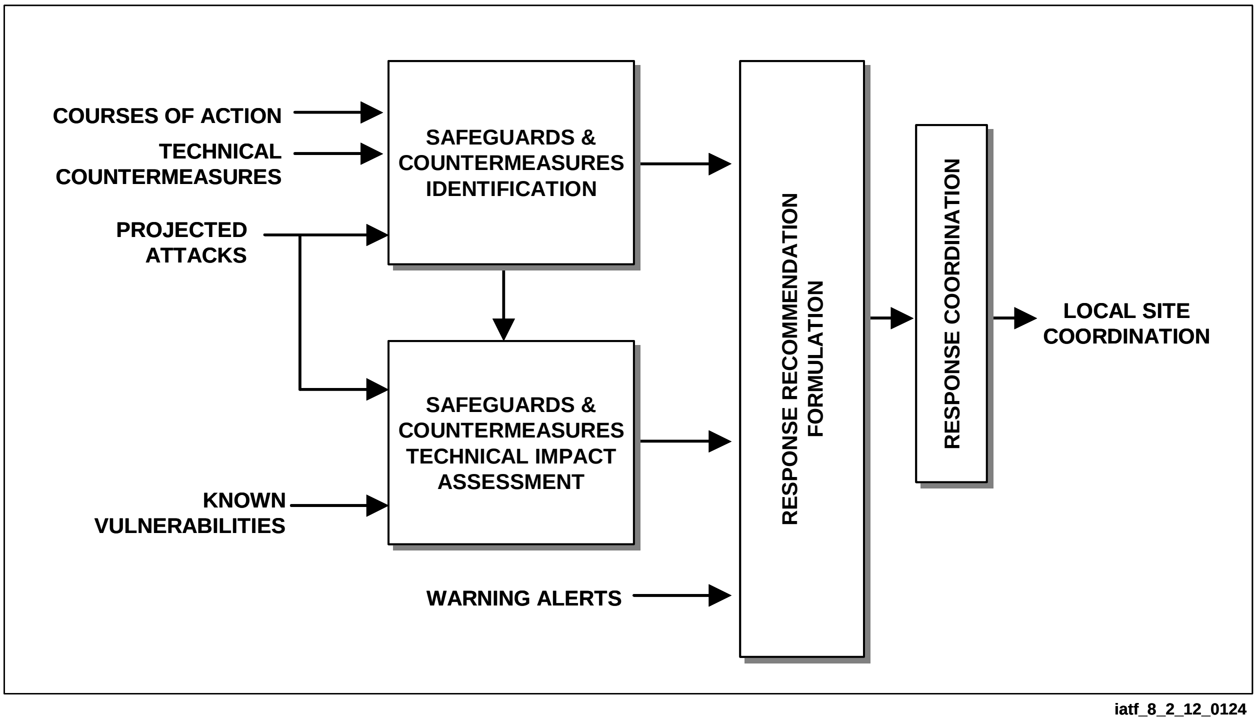

Figure 8.2-12. Functions to Support Response Coordination

Typically, local organizations and sites are in the best position to assess operational and mission impacts, based on projections of technical impacts to network services and system operations. Recommendations are formulated to assist local sites in the containment, damage assessment, eradication, and restoration to normal operational state. When appropriate, this also includes development or refinement of react mechanisms tailored to unique aspects of the ongoing attack.

8.2.4.9 Functions to Support Attack Investigation

This remaining set of functions focuses on analyzing how an attack was accomplished to provide feedback to improve existing (and future) protect, detect, and respond capabilities, ensuring that similar exploitations cannot occur. When appropriate, the investigation also structured to provide evidence of when prosecution of attackers is pursued. Typical functions associated with the attack investigation are shown in Figure 8.2-13. These functions are typically performed after the attack with extended time frames available for in-depth analyses. They can be considered in four basic groups or categories. The first is to establish and maintain a catalog of known vulnerabilities and the effects of known exploitations that provide a foundation for those analyses. These can include determining the effects of known attack sequences and potential modifications to those attack sequences. The second group, which is the primary focus for attack investigation, addresses characterization of the attack and attacker built from any available cyber evidence (e.g., audit logs, Transport Control Protocol [TCP] dumps).

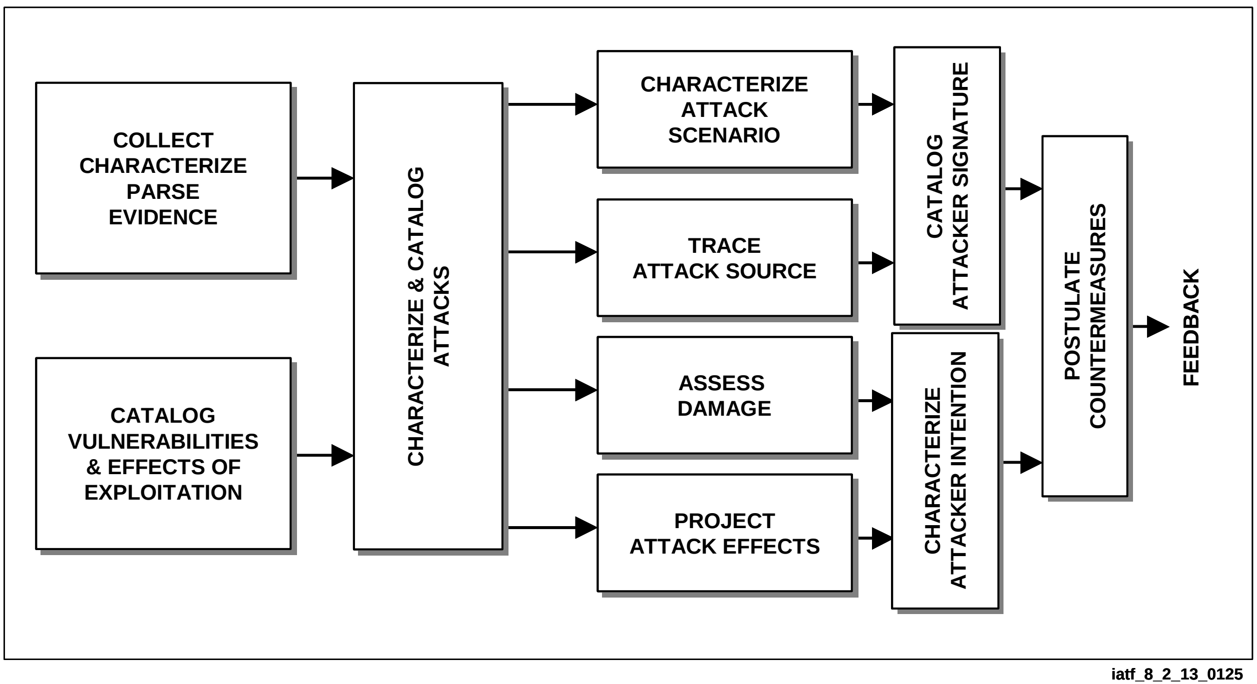

Figure 8.2-13. Functions to Support Attack Investigation

When required, this also provides evidence that could be used in subsequent prosecutions of attackers. The third establishes a set of attacker "signatures" (which could be thought of as a fingerprint file) that can be referenced when investigating future attacks. The remaining group focuses on developing and providing feedback for improving countermeasures and safeguards.

8.2.5 Relevant Detect and Respond Technologies

Cyber attack detection and response technologies (predominantly focused on intrusions) have emerged within the last several years as a result in large part of situations that stem from the worldwide interconnectivity created by the Internet. A computer-literate person can gain access into government and commercial internal networks via public routes using software hacking tools that can be easily downloaded from the Internet.

The previous section provided a perspective on the types of functions that are typical for various layers of a detect and respond infrastructure. This section provides guidance on technologies that are available to implement these functions and considerations for their selection and effective use. The section concludes with a reference model that provides an overall context for these technologies in a detect and respond infrastructure setting.

The Defense-in-Depth strategy and the overall Framework reinforce the close relationship of personnel, operations, and technology in realizing an effective information assurance (IA) posture. This cannot be emphasized too strongly across the detect and respond disciplines. When looking at the state of detect and respond technologies, it is clear that there are no "easy answers." Many of these technologies provide measurement (instrumentation) capabilities that must be interpreted by highly skilled analysts. Other technologies provide tools to support the analysis operations. Even the response technologies require well-trained and highly skilled operators to ensure that the response mitigates, rather than exacerbates, the effects of an incident or attack. Three major issues associated with effective technology deployment are—

- Where in the network they are deployed to ensure they address critical network resources
- How often they are used based on the operational concept of operation and availability of operators and analysts
- What skills the operators and analysts must have to make effective use of the results.

It cannot be over-emphasized that unlike protect technologies, detect and respond technologies do not in themselves offer any real protection. Rather, they enable the processes and functions that can mitigate the effects of an attack and restore the information systems and networks to an operational condition.

8.2.5.1 Technology Categories

Although commercial intrusion detection products have been available for several years, a number of recent and highly publicized hacking cases have created a renewed interest in the broader field of detect and respond technologies. Research by government, industry, and universities is ongoing to determine what constitutes an attack and how to detect and respond to an attack.

Today, most technologies tailored for detect and respond use provide information to an analyst, assist an analysis, or provide a means for responding based on the results of the analysis. Figure 8.2-14 shows the broad range of technologies that are addressed in this section of the Framework.

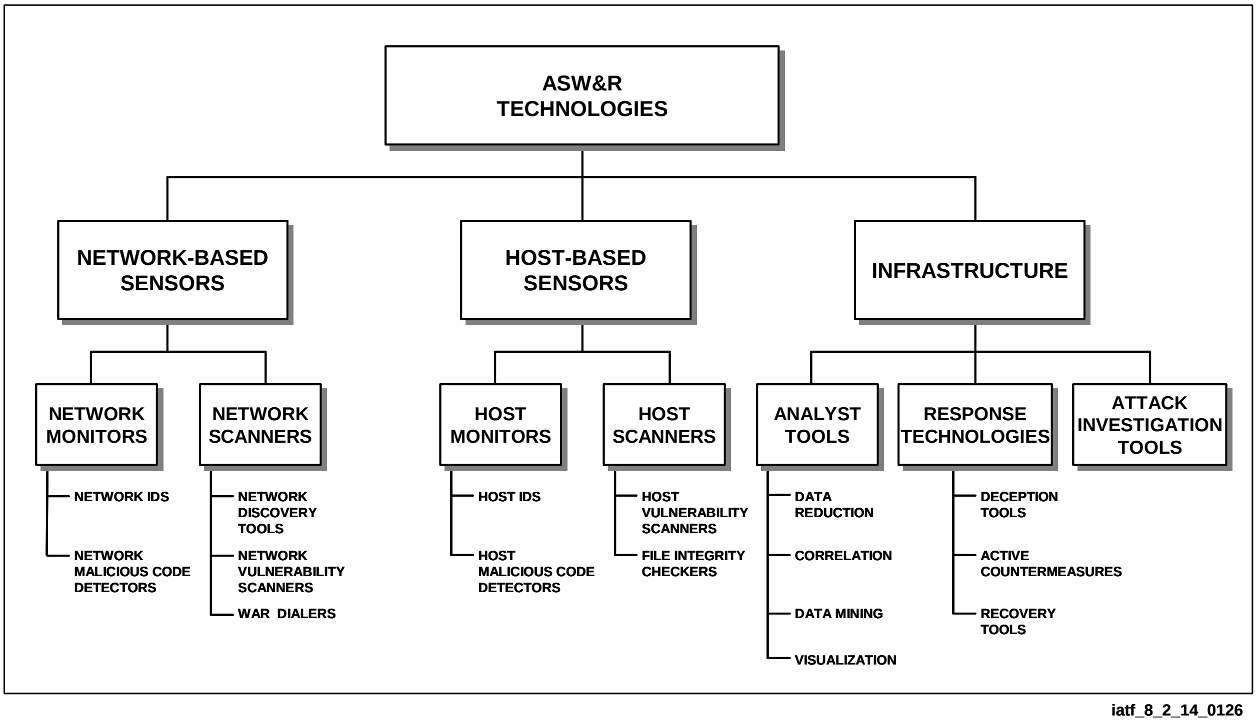

Figure 8.2-14. Detect and Respond Technologies

8.2.5.2 Monitoring and Scanning Technologies

It should be noted that monitoring and scanning technologies (characterized broadly as sensors) are covered in depth in other sections of the Framework. Specifically Sections 6.4, Network Monitoring within Enclave Boundaries and External Connections, and 6.5, Network Scanners within Enclave Boundaries, address network-based monitoring and scanners, respectively, while Section 7.2, Host-Based Detect and Respond Capabilities within Computing Environment, addresses host-based sensor technologies. This material is synopsized in this section to provide a context for the remaining technologies and to facilitate discussions of when and how to use these technologies in a synergistic fashion. Figure 8.2-15 identifies the general categories of these technologies.

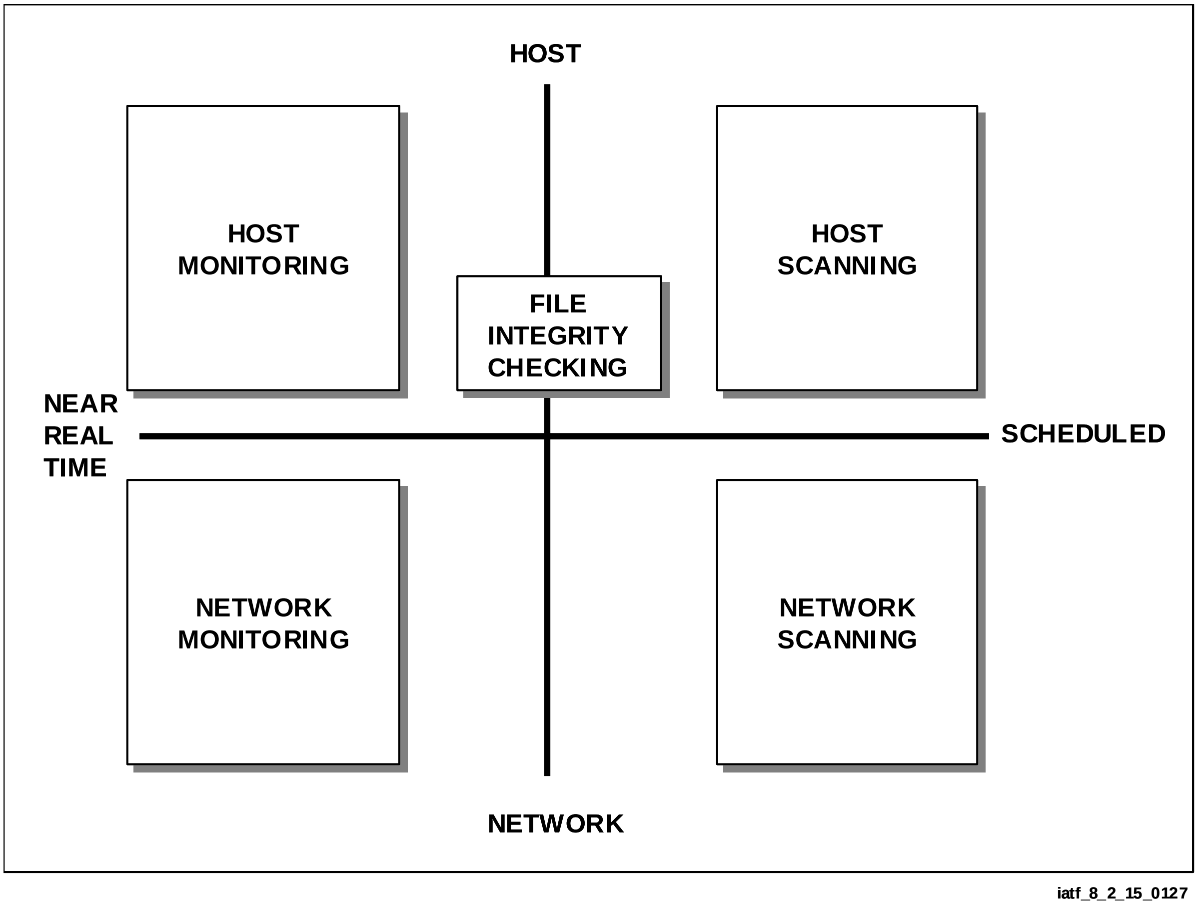

Figure 8.2-15. Sensor Technologies Grouping

Technology Overview

Network and host-based sensors provide alerts and supporting information to network operators and administrators that a vulnerable condition exists or an event has occurred within the enterprise and thus creates an opportunity for them to analyze and evaluate what actually transpired. This allows an appropriate action (as specified by the security policy for the organization) to be initiated. If the attack is detected in real time, it may be possible to mitigate the damage resulting from the attack. If detected after the attack is over, the logging features of the sensors may identify why the attack was successful so that exploitable weaknesses can be fortified.

Monitors

Network IDSs examine traffic on the wire in real time, examining packets looking for dangerous payloads or signs of abuse (e.g., malformed packets, incorrect source or destination addresses, and particular key words) to spot attacks before they reach their destination and do the damage. When suspicious activity is identified, a network-based IDS is capable of both raising alerts and terminating the offending connection. Some will also integrate with the firewall, automatically defining new rules to shut out the attacker in the future. As indicated in the earlier sections of the Framework, the high incidents of false positive detection make automated response mechanisms undesirable. Network-based IDSs typically operate on independent computers so there is no impact on the performance of mission systems. They are typically deployed one per network segment, because they are unable to see across switches and routers.

Host intrusion detection provides an agent that resides on each host to be monitored. The agent collects information reflecting the activity that occurs on a particular system. The monitor scans event logs, critical system files, and other auditable resources looking for unauthorized changes or suspicious patterns of activity. When anything out of the ordinary is noticed, alerts or Simple Network Management Protocol (SNMP) traps can be initiated automatically. The agent may also behave in a manner similar to the network-based IDS in that it will examine packets on the wire to compare against a database of known attacks—but in this case, it is restricted solely to packets targeted at the host machine. For this reason, host intrusion detection is ideal in a highly switched environment to protect specific critical servers, or for otherwise heavily loaded networks (where it may be difficult to protect the entire network). Some host-based IDSs also include a “personal firewall” capability to provide additional protection for the host machine. Unlike its network counterpart, host IDSs operate on mission-critical systems, and therefore, their performance impacts mission operations.

Malicious code detectors prevent and/or remove most types of malicious code. The use of malicious code scanning products with current virus definitions is crucial in preventing and detecting attacks by all types of malicious code. Malicious code detectors should be implemented across the enterprise. Defense against malicious code is only as good as its weakest link; if one system can be compromised, the entire enterprise is at risk. Centralized management for the AV capabilities with a common set of policies is strongly recommended.

Vulnerability Scanners

The Framework makes the distinction between scanners and the monitoring devices discussed above. Monitors typically operate in near real time and tend to measure the effectiveness of the network’s protection services in practice since they are subjected to actual exploitation attempts. Scanners, on the other hand, are preventative measures, typically operating periodically (or on demand) to examine systems for vulnerabilities that an adversary could exploit, evaluating effectiveness of the system infrastructure’s protection. Vulnerability scanners sometimes referred to as “risk assessment products” provide a number of known attacks with which network administrators can probe their network resources proactively. Scanners perform rigorous examinations of systems to locate known problems that represent security vulnerabilities.

Host-based scanners use an agent loaded on a system to examine a server or client. This examination can determine the potential system-level vulnerabilities that exist on a particular system based on known vulnerabilities in the operating systems. These technologies typically connect into a management console that can report on the status of all systems with agents across the network.

Network-based scanners examine a network and take inventory of all devices and components within the network infrastructure. These components, the network configuration, and the various versions of software controlling the network are examined and compared to a database of known vulnerabilities**.**

War dialers are a specialized type of network vulnerability scanner technology. Once identified, backdoors can be closed or some type of security plan created to preclude use of that particular point of entry. Along with a strong modem policy describing the need for modem registration and private branch exchange (PBX) controls, war dialer scanning can help an organization defend itself against such dangers. Use of this type of technology can help an enterprise identify vulnerable backdoors (e.g., unsecured modems across an enterprise) before an attack occurs.

File (software) integrity checkers are a specialized type of host scanner technology that verifies the integrity of files, detecting when files have been changed. As with the host vulnerability scanner technologies discussed above, these technologies tend to run off-line, and thus are not a protection mechanism. Typically they operate periodically, based on an event (e.g., file access) or on demand.

Considerations for Sensor Deployment and Operation

Deploying combinations of network and host-based sensors provides the best possible security by monitoring network-based traffic and host-specific exploitations directly on target workstations. This combination provides significant attack protection and facilitates policy enforcement for any size enterprise. Figure 8.2-16 identifies potential locations for their deployment.

When possible, it is recommended that the sensors be linked into the overall system and network management capabilities for an enterprise-wide solution. This eases individual sensor management, facilitates central reporting, and provides a more coherent perspective on the status of the enterprise overall.

Malicious code detectors should be implemented across the enterprise, on every system and network. Most of these technologies provide a means for sending responses or alerts at the server level, and some at the console level. It is always desirable to notify anyone that may have been infected that malicious code has been detected.

If scanners are deployed, it is important to consider what and when scans are performed. Otherwise, it is possible that mission-critical servers become busy responding to simulated attacks during times of peak demand. Assessment frequency is a factor of how often network changes are made as well as the security policy for the enterprise.

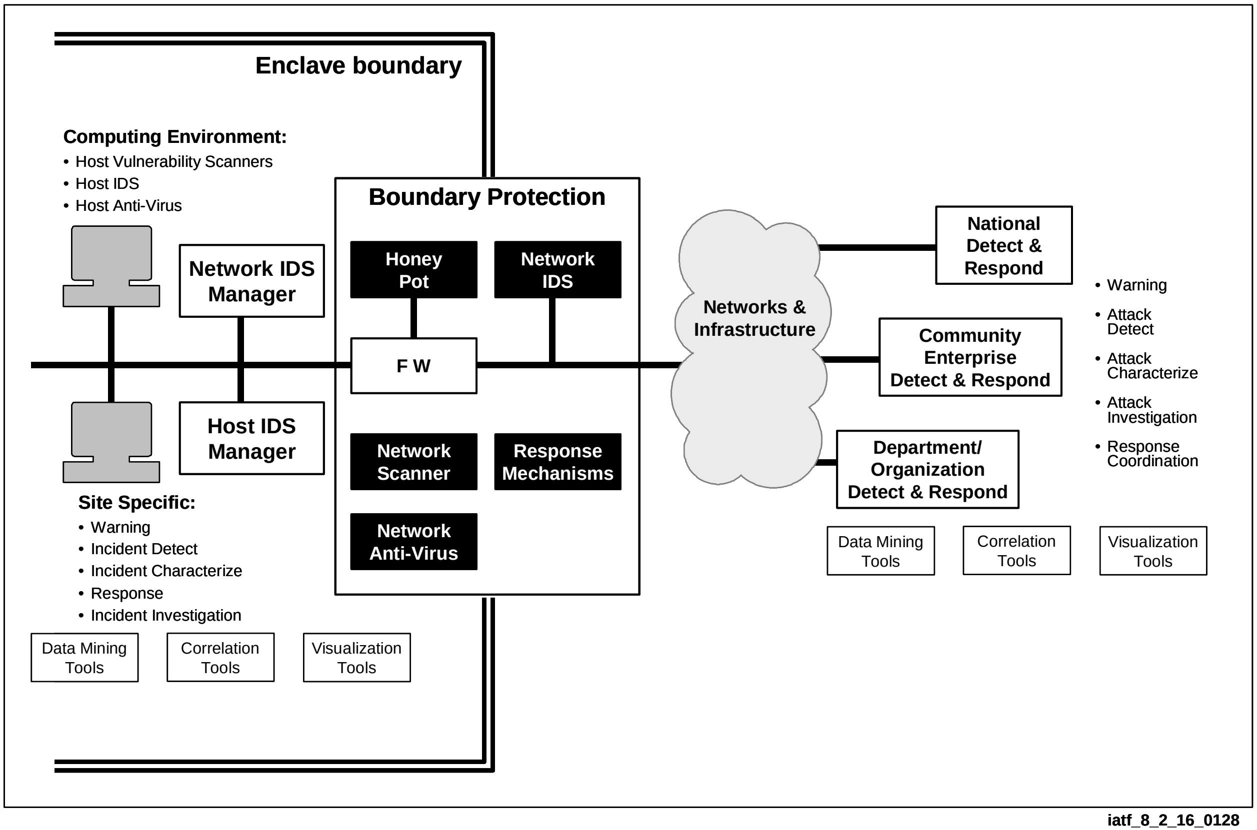

Figure 8.2-16. Possible Sensor Deployment Locations

The most important aspect to consider for integrity checker operation is deployment timing. To be their most effective, integrity checkers should be initialized on systems before they are placed into production and made generally accessible to their user communities. If they baseline monitored files and data structures any time after a system has "gone live," it is possible that the system has already become compromised and the integrity checker will miss changes that have already occurred.

8.2.5.3 Analyst Tools

Many intrusion detection and vulnerability scanning tools described above and in previous sections of the Framework come with their own rudimentary analysis tools. Some third-party vendors offer tools that will input security audit logs and intrusion event logs from some systems for further analysis, particularly if they have been generated in some standard format (e.g., open database connectivity). The interoperability standards for some of these formats (intrusion detection in particular) are still under development in standards bodies and government-sponsored activities such as the Defense Advanced Research Projects Agency (DARPA) Common Intrusion Detection Framework (CIDF) program, the Internet Engineering Task Force's (IETF) Intrusion Detection System Working Group, and the ISO SC27 standards group.

Technology Overview

While network and host sensor technologies have been developed specifically for detect and respond functionality, analyst tools have evolved from more general-purpose applications. Although basic tools and technologies exist, commercial analyst tools have not generally been tailored to this environment. We note that the government sector (e.g., the Intelligence Community) has developed a number of custom tools that more closely relate to this use; however, they are considered beyond the scope of the discussion in this Framework.

To support the analyst in performing the functions described in Section 8.2.4, Detect and Respond Functions, tools and techniques must be assembled that allow analysts to use all aspects of the information analysis technologies discussed below across the problem. The kind of tools required to do the "all source" type of analysis required by the detect and respond infrastructure are not currently available in the commercial sector, but any analyst tools (individually or in combination) must provide functions in the following areas:

Data Reduction. IDSs are notorious for generating large amounts of mostly superfluous information if not configured precisely. Even when well configured, their design is such that the system errs on the side of identifying, tagging, and reporting on all potential intrusion events. This data must be reduced to information of import before any additional analysis steps can be performed. Often, data reduction takes place incrementally during many of the analyst functions described in Section 8.2.4, Detect and Respond Functions. Models of "acceptable behavior" are typically used to reduce information. Local knowledge, such as configuration of the networking environment, knowledge of the application and systems in use across the network or enclave, and the expected traffic patterns of normal behavior, can all be used to reduce the mass of information generated by these systems to more manageable and germane levels.

Data Correlation. Correlation of events over a large set of data, even after data reduction techniques have been applied, to identify problems or determine if attacks are under way can be time-consuming and place extreme demands even on experienced operations staff. The larger the correlation environment, the more complex and detailed such correlations become. Often, operations staff cannot keep up with the increasing rates at which events are generated. Therefore, automated event management and correlation systems that can scale to large and complex environments are needed to accurately model and store the diagnostic knowledge possessed by operations staff. They must provide algorithms that analyze this knowledge in the context of the current system state to detect problems as they occur. Such systems must be able to input and correlate data from disparate sources, from intrusion detection event data to external alerts and intelligence databases. Generally, automated correlation tools determine relationships among data by implementing one or more of the following reasoning techniques: rule-based reasoning (RBR), model-based reasoning (MBR), state transition graphs (STG), codebooks, and case-based reasoning (CBR).

RBR techniques may not be well suited to larger, enterprise-wide environments but can work well in small domains, perhaps on the local level. Codebook reasoning is faster than rule-based reasoning given its streamlined encoding methodology and is better suited for larger enterprise environments. STG techniques are limited to correlated events in a single object and cannot determine when problems occur across related objects. MBR also does not function well in large domains, and CBR does not scale well because of the need for a general case library, which would be different for each enterprise/local environment. A scaled approach based on these techniques has yet to be developed.

Data Mining. Data mining refers to capabilities to drill down through a database and display information in a meaningful way. It is one segment of the broader knowledge discovery technology that addresses knowledge creation overall. Data mining technology and techniques can be applied to the analysis environment with the goal of turning information from all sources in the detect and respond infrastructure into the identification of hidden attacks, patterns of attacks, and prediction of attacks. Data mining technologies can potentially discover hidden predictive information in large data sets. They use knowledge discovery, pattern recognition, statistical data analysis, and database systems technology to automate the search for information in data sets. Data mining technologies collect and analyze information from multiple data sets and check them for data integrity. They provide a clearer resolution of the information, provide an understanding of attacks in progress, and predict patterns of attacks.

Some specific work is already under way at Columbia University, where researchers have defined and tested a data-mining framework for adaptively building intrusion detection models. Their work uses auditing programs to extract information to detail each network connection or host session. Then they apply data mining techniques, such as classification, meta-learning, association rules, and frequent episodes to learn rules that accurately capture the behavior of intrusions and normal activities. These rules can be used to build new detection models. While this is only part of the solution, it illustrates how data mining techniques are becoming an integral aspect of a more advanced detect and respond tools base.

Visualization. Data visualization cuts across all the aforementioned areas. Technologies must be employed that make use of simple, yet effective visualization techniques to assist the analyst through the various functions associated with the framework. The use of common metaphors and design elements provide the ability to visually process presented information effortlessly. Gestalt principles of proximity, continuity, similarity, symmetry or good form, and closure, as well as the introduction of appropriate perspective and relevant color, all significantly enhance the analysis functions.

Considerations for Their Selection, Deployment, and Operation

All the above factors must come together in a tool or series of technologies that provide to the analyst the ability to support the detect and respond infrastructure as described in Section 8.2.4, Detect and Respond Functions. Numerous tools exist that provide partial solutions, but there are still many challenges relating to common data export formats, the development of accepted reference models, and the problem of all-source data fusion that allow a focus on attacks versus incidents.

These technologies become of critical importance in the context of an overall enterprise management strategy, particularly as it pertains to detect and respond operations. Today, many event management functions are handled manually. Analysts and operators monitor and correlate events and handle identified problems (or potential problems). This manual processing does not scale to the growing speed, complexity, and size of many enterprise networks. Using these technologies, an enterprise management capability can accurately model and store diagnostic knowledge possessed by operations staff and provide algorithms that use this knowledge in the context of the current system state to monitor, detect, characterize, and react to events in an efficient and effective manner.

Essentially, all these technologies must—

- Operate on a common data/information format. Given the nature of the tools required and the information to be processed, some sort of data warehouse construct is probably the most viable approach.
- Provide different levels of functionality at different tiers of the framework. Some tool functionality, such as the requirement to integrate event information with intelligence data, will not be required at a local level but will be necessary at the organizational and national levels, particularly where coordinated attack determination analysis is under way.
- Provide seamless operator interfaces between technologies and a common, yet flexible, visualization approach.

There are no commercially available tools that provide all the necessary functions to satisfy the analysis needs within the detect and respond infrastructure. While there are fusion tools that have been developed within the government that provide functions similar to those needed for the detect and respond environment, they do not synergistically bring together various analysis technologies in a single packaging for this specific focused purpose. For the most part, they have evolved and have been tailored for specific community (e.g., warfare and intelligence) operations. In some cases, there are efforts under way to adapt them to the detect and respond environment; however, they have not reached the state of commercial technology offerings. Simple commercial off-the-shelf (COTS) approaches will undoubtedly require tailoring and integration efforts to build a cohesive shell or framework system around the various critical technologies.

8.2.5.4 Response Tools

There are two general classes of response tools considered within this Framework. One is a deception server, as discussed below. The second class of response tools, referred to as active countermeasures, focuses on implementing immediate mitigation actions to repel or redirect active attacks to minimize damage or reestablish and recover blocked or disabled services.

Deception Servers

These response tools provide capabilities for characterizing and refining information pertaining to attacks in progress or particular attackers either by redirecting or luring attackers into highly instrumented system infrastructures designed to closely audit all activities. These systems are typically called deception servers, although they are more commonly known as honey pots, fishbowls, and upon occasion, Venus flytraps.

Technology Overview

The concept behind deception servers is to present a "false" front, an instrumented server environment, with simulated well-know vulnerabilities (the honey pot construct) to lure attackers in with the promise of an easy score. These systems are designed and configured to emulate a production environment but are in reality set up to alert network administration and security staff while at the same time generating detailed activity logs of the attack or intrusion event. The system thoroughly measures and tracks the would-be intruder's activities.

While not a new idea, this is a relatively new class of product to be offered commercially. These products are capable of simulating a range of different network servers and devices to act as an attractive decoy for the would-be attacker. While the attacker concentrates on the decoy services, the honey pot collects as much evidence as it can while it is alerting the administrator.

When an incident is detected it is the organization's choice to terminate the connection immediately or to continue to allow the attacker to explore the façade system. If the connection is terminated, the attackers know that they have been detected and may try a different approach or to attack a different organization with the same attack. If allowed to continue unchallenged within the deception environment, information about the attacker can be gained. This information can be recorded and used by law enforcement officials to apprehend the attacker and take suitable legal action.

Deception servers can be useful only if the environment being protected has sufficient resources to use them once they are deployed.

Considerations for Selection Unique to the Deception Server Environment

Besides the usual criteria for selection of any software package or technology to be used within the Framework, such as supportability, dependability, clarity of user interface and documentation, ease-of-use and the like, there are a few fairly unique aspects to consider. There are a number of considerations that should be taken into account when choosing a deception server product for deployment.

Platform and Emulation Operating System. The most important factor to consider is platform support. The system should either run on the same type of platform that is commonly used in the environment it will be protecting, or emulate the operating system that is running on the true production systems that surround it. Depending on the target environment, Windows NT and various versions of UNIX should be supported. Some products will even attempt to emulate network appliance services such as Cisco Internet Operating System (IOS).

Commercial Product versus "Home Grown". There are numerous documents available in the community that describe how to configure a deception server from base operating system installations. This could be considered as a cost savings option, particularly if there are operating system support personnel available. However, it may be much more efficient to simply use one of the available products "out of the box."

Emulation Level. Some deception servers attempt to emulate more commonly offered network services while others emulate the application level. The closer the emulation to the true implementation, the more likely the ruse will work without alerting the attacker to the deception. One available technology actually makes a copy of your production system environment, securing it and instrumenting it on a second hardware platform for deployment as a deception server. Those systems that only emulate at an application level are susceptible to network-level operating system (OS) identification tools, such as the commonly used Nmap. The level of deception required depends upon how high the risk factors are for the environment and the probability of threats coming from highly sophisticated attackers. For environments with few resources, easily deployed, commercially available emulation packages should suffice. However, for the best coverage, a full-blown dedicated system that imitates the production environment in every way will provide the best protection possible.

Reporting and Logging. Of course, the depth and breadth of logging are important, particularly based on what the true operational goals of the deception server are. If the goal is to simply be alerted to the fact that an intrusion is under way and provide some level of data to assist in the foiling of the intrusion and recovery, the level of audit and reporting need not be particularly high. However, if the goal is to provide sufficient evidence to law enforcement officials to trace and potentially prosecute an attacker, a higher level of audit, reporting, and supporting documentation are required.

Considerations for Deployment and Operation

There are a number of considerations for deployment and operation of deception servers.

Placement on the Network and Redirection. Several methods exist for placing deception servers into a network infrastructure and ensuring attackers go after it. For example, one can either set up boundary routers or firewalls to redirect nonproduction services (e.g., File Transfer Protocol [FTP] or Telnet) to the deception servers rather than to just not support them, and then route normal services, such as HTTP, to production systems. The drawback, of course, is that if attacks take place using production services, the deception server provides no added value. Another approach is to place a deception server at the same logical network level as production servers and have it emulate full production services, so it can become targeted in attack "sweeps."

Legal Issues. Little or no legal precedence has been established for deception servers. If deception servers are deployed, some potential liabilities could be experienced. It would be wise to post the same restricted use notifications that are found on the enterprise's true production systems. Additionally, be prepared that if the deception server is compromised and then subsequently used as a stepping off point for attacks elsewhere, the organization that deployed the deception server could be found culpable, more so than if their normal production servers were compromised despite due diligence efforts. It should be kept in mind that deception servers are detection tools and should be treated as such, and unless the deploying organization is a law enforcement agency, unfair entrapment charges cannot really be made successfully.

Active Countermeasures and Recovery Tools

Active countermeasures and recovery tools focus on terminating the intrusion or attack and restoring affected services or lost data as soon as possible. Recovery may also include initial (technical) damage assessment tools that ascertain the extent of the damage inflicted during the intrusion or attack. These should be differentiated from attack investigation tools, which are used to gather information about intrusions with the intent, among other activities, to trace, locate, apprehend, and prosecute intruders and attackers (addressed in subsequent sections).

Technology Overview

Reconfiguration, Containment, and Disconnection Technologies. There are numerous approaches to initiating active countermeasures that serve to halt or block attacks that are discovered against an environment. Typically, there are no tools one can acquire that stand alone and are used to repel attacks. Most countermeasures come bundled with IDSs. They provide either a standalone capability (e.g., the ability to send TCP disconnects to certain active connections determined to be the source of attacks), have programmed interfaces to network equipment (switches, hubs, routers, and firewalls) so certain connections can be cleared or blocked at the network level, or allow new filtering rules to be instituted based on addressing or protocols associated with the attack. Many tools allow the creation of precanned scripts that can be executed causing dynamic reconfigurations across the enterprise.

Additionally, some host-based tools provide the ability to interface with the host operating system to allow quick disabling of accounts that are being used as launch points for attacks. Dynamic access control modifications are also possible. All these tools should be focused on minimizing the period in which the attack takes place, and consequently minimizing the damage, either from the original attack or as the intruders attempt to cover their tracks as they back out.

These tools (or more appropriately features of available intrusion detection tools) must be chosen carefully and their use within the secure infrastructure planned accordingly. Each of the various attack mitigation features should be thoroughly tested to ensure that they do not wreak more havoc on the enterprise than the original attack. Some tools allow the automatic institution of countermeasures. It is recommended that automatic “shunning” not be implemented until all scenarios are tested and sufficient operational experience in the particular environment indicates the risks are minimal.

Recovery Tools. Damage assessment and recovery tools include disk repair and recovery tools as well as operating system specific tools that are able to make repairs to OS-specific data structures on the system (e.g., the Windows registry). It is important to prepare these tools ahead of time, in anticipation of having to recover from attacks, because no protection features are foolproof.

Backup recovery tools are an important component of this part of the framework. Each set of tools must be chosen to work with the particular platforms and information system applications running within the enterprise. Preevent planning and rehearsals should be conducted to ensure that the tools are configured appropriately and operations personnel are sufficiently trained. Processes and procedures for proper backup execution, testing, and the selection of the appropriate periodicity to execute backups are all critical factors in the preplanning phases of recovery operations. Some of the file integrity checking tools addressed in Section 7.2.4, Host Scanners—File Integrity Checkers, can also be used in the recovery process, determining which files may have been corrupted during the attack and may have to be restored from protected media. Besides the technology, appropriate planning is an absolute necessity as part of any response capability.

8.2.5.5 Attack Investigation Tools

Also referred to as computer forensics tools, attack investigation tools, and computer forensics science in general, focuses on acquiring, preserving, retrieving, and presenting information associated with illegal intrusion activities. Three roles of a computer within a criminal context have been identified. The first is where the computer is a target of an attack or intrusion. The second is where a computer is the used as an instrument of an attack (a hacker's computer, for instance). The third is where a computer may be a repository for information pertaining to the commission of a crime, containing databases, images, etc.

In the context of the detect and respond infrastructure, attack investigation or forensics tools consider the first and second roles. The first, where the computer is the subject of the attack, and the second, where a third-party computer is attacked and usurped, then used in subsequent attacks on other systems. The aspect of seized computers being examined for their role in criminal activities, whether as a tool or as a repository, is beyond the scope of this section of the Framework.

Technology Overview

There are three general phases to any computer forensics process: acquisition, examination, and utilization, and consequently different tools for each. In the acquisition phase, information must be acquired from the systems that have been intruded upon and/or attacked in such a way that all the information on the system is captured. In situations where criminal prosecutions are a goal of the investigation, the information must be collected and maintained consistent with rules of evidence. In the examination phase, appropriate tools must be used to analyze the information on the system with the intent of attempt to ascertain such facts as—

- How the attack was achieved (i.e., what vulnerability, technical or procedural, was exploited).
- What information the intruder may have left behind to implicate himself or herself (e.g., trace logs, malicious code or Trojan Horse software, trademark methods, system damage).

- What the intent of the intruder was (e.g., exploration/curiosity, malicious damage, information theft, denial of service, service theft).

Finally, the utilization phase of the forensics process allows for the creation of formal reports, the certification of the chain of custody thread, and all other aspects that then allow the pursuit of a criminal investigation leading to a potential prosecution.

Most standard computer forensics tools focus on the preservation of evidence, the analysis of information for criminal activity, and then the final packaging for prosecution. In the detect and respond infrastructure, while many of these tools have applicability, additional analysis tools that focus on log and event analysis are also important. Of particular importance are those logs and events from secondary systems such as routers and firewalls and not necessarily just the pilfered target system itself. However, in all cases, rules of evidence must be followed to support successful prosecutions.

When a situation arises in the detect and respond environment where attack analysis is intended to potentially lead to criminal prosecution, acquisition tools that capture and preserve the evidentiary trail of information must be used instead of simple log or event information capture and copying. Tools that make exact, certifiable copies of information and often entire disk images must be deployed.

For analysis, tools that not only attempt to recover lost or deleted information (an intruder covering his/her “tracks”) must be deployed, but tools that analyze log events and audit information to build a profile of how an intrusion progressed must also be applied. If necessary, tools that can analyze down to individual TCP/Internet Protocol (TCP/IP) segments and datagrams (TCPdump) must be used along side the more traditional computer forensics tools.

Finally, tools that generate reports, document the chain of custody, and just generally provide additional efficiency, fill out the third phase, utilization.

Considerations for Selection and Operations

There are a number of factors associated with attack analysis that should be considered when pulling together a stable of appropriate tools.

Ease of Use and Integration. A clean, robust user interface, particularly in the complicated analysis phases of an investigation, is critical. Many tools handle all aspects of attack investigation (acquisition, analysis, and utilization) in complete packages, most focused on computer crime scene investigation. It is important to consider if these all-in-one packages adapt easily to the operational environment in question. Also, in most cases, a long, drawn-out investigation will have prohibitive impact on operations. The speed with which information can be collected for later analysis is critical.

Preservation of Evidence. The tools must preserve evidence appropriately, per acceptable law enforcement or prosecutorial standards. The disk copying or information copying tools must function in such a way as to ensure a perfect copy is preserved. The available tradeoffs between speed and copy perfection (full image versus information copy) must be determined.

Flexibility. The tools must be able to collect, preserve and analyze information from the systems deployed in the local environment.

Operational Approach. The available tools are still in focused mostly on single activities, such as information capture or disk imaging, log analysis, the discovery of deleted files or hidden information. Consequently, particularly in a detect and respond situation, a well-composed investigative framework must be established ahead of time to provide the context for the implementation of the tools. The functions during an investigation are described in Section 8.2.4.9, Functions to Support Attack Investigation, but the next level of detail appropriate to the particular environment in question, such as operations personnel availability, budget, local and/or national policies on how long systems can remain off-line for investigation purposes, etc., all must drive the particular tool acquisitions.

8.2.5.6 Related Detect and Respond Operational Considerations

While there are a number of technologies available to support various aspects of detect and respond, there are also important considerations that deal with their selection, deployment, and operation. Some of these are discussed below.

Independent Testing of Technologies

Another factor slowing the development of these technologies is the lack of adequate testing and product certification facilities. Large-scale testbeds are needed to test these systems using real-world simulations and to develop metrics, verification procedures, and standard test-case scenarios. There is a real need for independent laboratories to evaluate and certify products, providing unbiased and accurate evaluations of relevant technologies that can be made available to network security customers.

The National Security Telecommunications and Information Systems Security Policy (NSTISSP) No. 11 provides the national policy that governs the acquisition of IA and IA-enabled information technology products for national security telecommunications and information systems. This policy mandates that effective January 2001 preference be given to products that are in compliance with one of the following:

- International Common Criteria for Information Security Technology Evaluation Mutual Recognition Arrangement.
- NSA/National Institute of Standards and Technology (NIST) National Information Assurance Partnership (NIAP).
- NIST Federal Information Processing Standard (FIPS) validation program.

After January 2002, this requirement is mandated. DoD Chief Information Officer (CIO) Guidance and Policy Memorandum No. 6-8510, *Guidance and Policy for Department of Defense Global Information Grid Information Assurance* references this same NSTISSP No. 11 as an acquisition policy for the Department.

The International Common Criteria and NIAP initiatives base product evaluations against Common Criteria Protection Profiles. NSA and NIST are working to develop a comprehensive set of protection profiles for use by these initiatives.

System Backup

There are two main strategies to follow when performing a system backup: one for the workstation level and the other for the network level.

Workstation Strategy

The best backup strategy for workstations is to back up often. If the workstation is running the Windows OS, there are some simple backup tools already provided. There are also several utilities and programs available from reputable companies to aid users in performing backups. The following features can make backup chores more bearable: incremental backup, unattended scheduling, and easy, simple restoration. Incremental backup saves changes made since the most recent full or incremental backup. This is important because users who do not want to wait to back up a system can use incremental backup as a substitute for a lengthy full backup. Scheduling uses software automation to execute backup chores without the need for personal interaction. While the user must select and put in place a backup media, the user does not need to be present for the actual backup. Zip© drives and small tape drives are also cost-effective solutions used to back up workstation data.

Network Strategy

The best backup strategy for networks is an approach that combines several features to save time and effort and still ensure complete backups. Execute full backups often. Since backups take up network, server, and/or workstation resources, it is best to run full backups when none is working. Also, open files are skipped during backup and do not get backed up at all until some future time when the file is closed and not being used. Having few to no users holding files open will ensure the greatest backup saturation possible. Full backups are most efficiently executed in the evenings. Store the full backup tape off-site. On each of the remaining workdays of the week, using a separate tape for each day, run an incremental backup and store it off-site, too. The last full backup of the month should be permanently moved off-site and held for archival purposes. If a network is attacked by malicious code, these backup techniques will ensure data integrity and allow all systems to be recovered.

Security Awareness Training

Security awareness is usually a first line of defense for an organization. Organizations should implement a security awareness training program sanctioned by a recognized information systems security authority such as NIST. An acceptable security program should be able to inform users about the threats of e-mail attachments, simple physical security, and protection of authentication mechanisms. The threats are much more numerous than these examples but statistical information indicates most users know very little about these threats.

Configuration

Proper system administration is one of the best mechanisms to limit the number of vulnerabilities that can be exploited. CERT and other organizations publish vulnerabilities and fixes for those vulnerabilities. Every organization should be aware of the latest security patches and fixes for their equipment.

Privacy Concerns

Organizations may own the intellectual property of employees and may also legally restrict computer activities to only those approved by management. A common practice is to present this warning to all computer users as part of the normal login message. This does not mean that all managers in an enterprise own all of the transactions of all of the employees. Especially unclear is how to handle the conflict that arises between privacy and monitoring. Use of IDSs and system-monitoring tools requires caution. Legal issues pose a potential problem to the deployment and use of detect and respond technologies. As noted in NTIB#1, legal and regulatory issues are very complex and the "legal system has not yet made authoritative judgments on the issues." The report illustrates the conflicting views on the subject noting that "intrusion detection systems are sometime viewed as intrusive themselves, and . . . the position is taken that all information systems are subject to arbitrary monitoring at any time."[3]

Sniffers that search for key words in messages (e.g., "attack," "weakness," or "confidentiality") as a standard set of watchwords may find key words used in an appropriate manner depending on the type of correspondence. Audit trail reports may contain full command strings (including parameters). The results of an analyst's investigation of traffic patterns or traffic content within or interfacing to an enterprise (either in response to a possible intrusion or during an investigation following an attack) could be considered an unwarranted invasion of privacy. Activating and directing a potential adversary to a honey pot (deception server) raises privacy issues as well. It is important to refer privacy concerns to the appropriate legal and policy organizations for the enterprise prior to deployment and use of these technologies.

---

3 "National INFOSEC Technical Baseline—Intrusion Detection and Response," Lawrence Livermore National Laboratory and Sandia National Laboratories, December 1996, as reported in Network Intrusion Detection and Response, a Technology Forecast, by William L. Cameron, AlliedSignal Technical Services Corporation, August 1998.

8.2.5.7 Technology Reference Model

As discussed earlier in this section of the Framework, the detect and respond infrastructure is hierarchical by its nature. There is a tight coupling between the physical structures (of the local computing environment, enclave boundary, and system infrastructures), the processes that need to be performed, and the technologies that are available to realize those processes at each layer of the hierarchy. A technology reference model for this system infrastructure highlighting these relationships is provided in Figure 8.2-17.

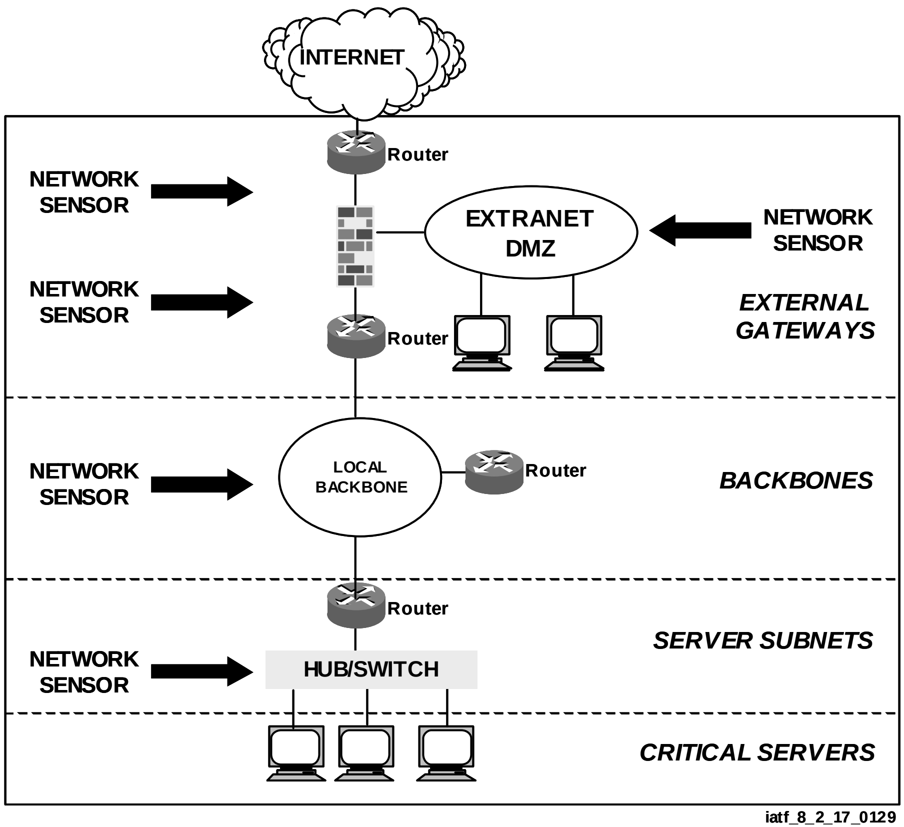

Figure 8.2-17. Detect and Respond Technology Reference Model

The shaded areas of the figure represent a typical local environment (computing environment and enclave boundary). As discussed in earlier sections, the local environment is the natural location for host and network-based sensors (e.g., IDSs and vulnerability scanners). If detect and respond technologies (e.g., honey pots) are used, they are also located at this level of the hierarchy.

The processing above the sensors can be placed at every level of the hierarchy. Local environments have the option of deploying any and all aspects of processing and analysis, usually focused for their specific operations. Similar structures may also be available to focus at organizational, enterprise, and national levels. There is a decision making capability needed at each level to interpret the operational implications of current situations and provide direction on courses of actions. This is typically performed with some collaboration at levels higher and lower as appropriate.

The network infrastructures that typically connect local environments together also provide the basic connectivity of these environments to various elements of the detect and respond infrastructure. This connectivity is needed to provide reporting up the hierarchy and information associated with response coordination back down.

The very nature of the reference model highlights the importance of selecting technologies that can interoperate with each other across the overall detect and respond infrastructure. Although not shown, to realize a system infrastructure that can deal with an appreciable sized enterprise, that integration should extend into the system and network management infrastructures as well.

8.2.6 For More Information

The list of reference materials used in preparing this section provides an excellent base of knowledge from which to draw on relevant technologies. There are a number of additional sources of information. This section of the Framework focuses on on-line sources because they tend to offer up-to-date information. These include the following:

IA Technology Framework Executive Summaries

An important segment of the IATF is a series of executive summaries that are intended to provide summary implementation guidance for specific case situations. These offer important perspectives on the application of specific technologies to realistic operational environments. These are still being formulated and will be posted on the IATF Web site http://www.iatf.net/ as they become available.

Protection Profiles

The International Common Criteria and NIAP initiatives base product evaluations against Common Criteria Protection Profiles. NSA and NIST are working to develop a comprehensive set of protection profiles for use by these initiatives. An overview of these initiatives, copies of the protection profiles, and status of various products that have been evaluated are available at the NIST Web site http://niap.nist.gov/.

8.2.6.1 Independent Third-Part Reviewers of Relevant Vendor Technologies

- ICSA Net Security Page, www.icsa.net
- Talisker's Intrusion Detection Systems, www.networkintrusion.co.uk/
- Network Computing—The Technology Solution Center, www.nwc.com/1023/1023f12.html

- Paper on CMDS Enterprise 4.02, http://www.Intrusion.com/Products/enterprise.shtml (ODS Networks has changed its name to Intrusion.com)
- PC Week On-Line, www.zdnet.com/pcweek/reviews/0810/10sec.html

8.2.6.2 Overview of Relevant Research Activities

- Coast Homepage—Perdue University, www.cs.purdue.edu/coast
- UC Davis, seclab.cs.ucdavis.edu/

8.2.6.3 Overview of Selected Network Monitor Vendor Technologies

- Symantec Corporation, http://www.symantec.com
- Cai.net, http://www.cai.net/
- Cisco Connection Online, www.cisco.com
- CyberSafe Corporation, www.cybersafe.com
- Internet Security Systems, www.iss.net
- Network ICE, www.networkice.com

References

1. Information Assurance Technical Framework (IATF), http://www.iatf.net.

2. National Institute of Standards and Technology, http://niap.nist.gov/.

Additional References

a. Amoroso, Edward, *Intrusion Detection*. Intrusion.Net Books. 1999.

b. Symantec Corporation. *Intruder Alert 3.5 IDS Review Guide*. May 2000.

c. Symantec Corporation. *Everything You Need to Know About Intrusion Detection*. 1999.

d. Balasubramaniyan, J. S., et al. *An Architecture for Intrusion Detection Using Autonomous Agents*. COAST Technical Report. 11 June 1998.

e. Concurrent Technologies Corporation. *Attack Sensing, Warning, and Response (ASW&R) Trade Study Report Intrusion Detection System*. Report No. 0017-UU-TE-000621. April 14, 2000.

f. Concurrent Technologies Corporation. *Attack Sensing, Warning, and Response (ASW&R) Baseline Tool Assessment Task Anti-Virus Trade Study Report*. Report No. 0017-UU-TE-000623. April 13, 2000.

g. Department of Defense (DoD) Chief Information Officer (CIO) Guidance and Policy Memorandum No. 6-8510, *Guidance and Policy for Department of Defense Global Information Grid Information Assurance*.

h. Escamilla, Terry. *Intrusion Detection, Network Security Beyond the Firewall*. Wiley Computer Publishing. 1998.

i. Graham, Robert. “New Security Trends for Open Networks.” *SC Magazine*. October 1999.

j. Information Assurance Technology Analysis Center (IATAC). *Tools Report on Intrusion Detection*. Defense Technical Information Center. December 1999.

k. Information Assurance Technology Analysis Center (IATAC). *Tools Report on Vulnerability Analysis Information*. Defense Technical Information Center. March 15, 2000.

l. “Intrusion Detection.” *SC Magazine*. June 2000.

m. Maes, V. “How I Chose an IDS.” *Information Security Magazine*. Volume 2, Number 9. September 1999.

n. National Security Telecommunications and Information Systems Security Policy (NSTISSP) No. 11. *National Policy Governing the Acquisition of Information Assurance (IA) and IA-Enabled Information Technology (IT) Products*. January 2000.

o. Northcutt, Stephen. *Network Intrusion Detection, An Analyst's Handbook*. New Riders Publishing. 1999.

p. Schneider, Sondra, et al. "Life After IDS." *Information Security Magazine*. Volume 2, Number 9. September 1999.

q. Snapp, Steven R., et al. *A System for Distributed Intrusion Detection*. IEEE CH2961-1/91/0000/0170. 1999.

r. Ulsch, Macdonnell and Joseph Judge. "Bitter-Suite Security." *Information Security Magazine*. Volume 2, Number 1. January 1999.

This page intentionally left blank.